

(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2020-526820  
(P2020-526820A)

(43) 公表日 令和2年8月31日(2020.8.31)

(51) Int.Cl.	F I	テーマコード (参考)
<b>G06F 21/64 (2013.01)</b>	G06F 21/64	
<b>G06F 21/33 (2013.01)</b>	G06F 21/33	

審査請求 未請求 予備審査請求 未請求 (全 51 頁)

<p>(21) 出願番号 特願2019-571965 (P2019-571965)</p> <p>(86) (22) 出願日 平成30年6月28日 (2018. 6. 28)</p> <p>(85) 翻訳文提出日 令和2年2月19日 (2020. 2. 19)</p> <p>(86) 国際出願番号 PCT/US2018/040105</p> <p>(87) 国際公開番号 W02019/006177</p> <p>(87) 国際公開日 平成31年1月3日 (2019. 1. 3)</p> <p>(31) 優先権主張番号 62/527, 479</p> <p>(32) 優先日 平成29年6月30日 (2017. 6. 30)</p> <p>(33) 優先権主張国・地域又は機関 米国 (US)</p> <p>(31) 優先権主張番号 15/829, 684</p> <p>(32) 優先日 平成29年12月1日 (2017. 12. 1)</p> <p>(33) 優先権主張国・地域又は機関 米国 (US)</p>	<p>(71) 出願人 502303739 オラクル・インターナショナル・コーポレーション アメリカ合衆国カリフォルニア州94065 レッドウッド・シティー, オラクル・パークウェイ500</p> <p>(74) 代理人 110001195 特許業務法人深見特許事務所</p> <p>(72) 発明者 ダンカン, ミルズ イギリス, エス・オウ・22 5・ディ・エス ウィンチェスター, コーンズ・クローズ, 30</p>
---	---

最終頁に続く

(54) 【発明の名称】 分散型台帳を用いて公共のソフトウェアコンポーネント・エコシステムを管理するためのシステムおよび方法

(57) 【要約】

実施形態は、カタログインスタンスと称されるエコシステムフルノードによって維持される、たとえば、部分的に分散型ブロックチェーンとして実現されるような、分散型台帳を用いるソフトウェアコンポーネント・エコシステムの実現を容易にするシステムおよび方法を含む。さまざまな登録トランザクション（たとえば、デベロッパ登録、ネームスペース登録、使用登録など）は、1つ以上のカタログインスタンスによって分散型ブロックチェーンに追加され得る。ブロックをブロックチェーンに追加するステップは、トレンジングトランザクションのツリーハッシュを用いて新しいブロックのためのブロックハッシュを生成するステップを含み得る。新しいブロックはまた、直前のブロックのハッシュを含んでいるかまたは組込んでいる。

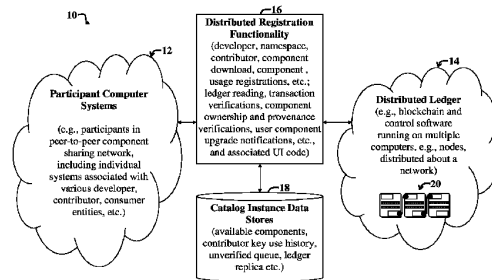


FIG. 1

**【特許請求の範囲】****【請求項 1】**

1つ以上のプロセッサによって実行すべき論理を含む1つ以上の有形な媒体であって、前記論理は、実行されると、以下の複数の動作を実行することによってソフトウェアコンポーネント・エコシステムを実現するように動作可能であり、前記以下の複数の動作は、

ユーザ入力デバイスから1つ以上の信号を受取って、デベロッパエンティティの識別名を規定する動作と、

ユーザ入力デバイスから1つ以上の信号を受取って、前記デベロッパエンティティが前記識別名を登録することを認可されていることを確認するのに使用可能な安全な検証メカニズムを規定する動作と、

前記安全な検証メカニズムと関連付けてデベロッパエンティティ名を含むカタログエントリを記録する動作と、

前記カタログエントリと関連付けてトランザクションを実行して、ソフトウェアコンポーネントを転送する動作と、

前記カタログエントリを用いて、前記トランザクションを検証する動作とを含み、これにより、前記トランザクションを特徴付けるデータがカタログにコミットされる、1つ以上の有形な媒体。

**【請求項 2】**

前記安全な検証メカニズムは公開鍵およびデジタル署名を含む、請求項 1 に記載の1つ以上の有形な記憶媒体。

**【請求項 3】**

前記ソフトウェアコンポーネント・エコシステムにおける複数のノードに前記カタログエントリを伝搬する動作をさらに含む、請求項 2 に記載の1つ以上の有形な記憶媒体。

**【請求項 4】**

前記複数のノードの各々は、分散型ブロックチェーン台帳のレプリカまたは近似レプリカを維持する前記複数のノードと協調して前記分散型ブロックチェーン台帳を実現するためのコードを含むカタログインスタンスを表わす、請求項 3 に記載の1つ以上の有形な記憶媒体。

**【請求項 5】**

前記エントリにネームスペースを割付ける動作をさらに含む、請求項 2 に記載の1つ以上の有形な記憶媒体。

**【請求項 6】**

複数の相互通信カタログインスタンスを含む、請求項 1 に記載の分散型ピア・ツー・ピア・コンポーネント・エコシステムのコンポーネントカタログのインスタンスを用いて、

前記デベロッパエンティティ名に関するデベロッパエンティティ登録情報を受取るステップと、

前記デベロッパエンティティ登録情報を検証して、前記デベロッパエンティティ登録情報を、デベロッパエンティティ登録の記録またはブロックとして、前記複数の相互通信カタログインスタンスの各々にアクセス可能であるブロックチェーンとして実現される分散型データベースにコミットすることにより、前記分散型ピア・ツー・ピア・コンポーネント・エコシステムのフルノードすべてに対して前記デベロッパエンティティ登録情報を伝搬するステップとを実行する、請求項 1 に記載の1つ以上の有形な記憶媒体。

**【請求項 7】**

前記受取るステップは、

前記デベロッパエンティティ名と、登録が実行されているカタログインスタンスに関連付けられた公開鍵と、前記デベロッパエンティティ名に関連付けられたデジタル署名とをデベロッパエンティティにより供給するステップの後に行なわれ、前記デジタル署名は、前記複数の相互通信カタログインスタンスのカタログインスタンスで、デベロッパエンティティアイデンティティを確認するのに使用可能である、請求項 6 に記載の1つ以上の有

10

20

30

40

50

形な記憶媒体。

【請求項 8】

前記分散型ピア・ツー・ピア・コンポーネント・エコシステムの前記フルノードは、前記複数の相互通信カタログインスタンスを含む、請求項 6 に記載の 1 つ以上の有形な記憶媒体。

【請求項 9】

ネームスペース登録プロセス中の 1 つ以上の供給されたネームスペースをデベロッパ登録情報に関連付けることによって、

前記デベロッパエンティティ登録情報を伝搬した後、前記ネームスペース登録プロセスの開始および実現に回答して、1 つ以上のネームスペースをデベロッパエンティティに割付けることをさらに含む、請求項 6 に記載の 1 つ以上の有形な記憶媒体。

10

【請求項 10】

いくつかの記録が前記ブロックチェーンの一連の記録にコミットされる後まで、デベロッパエンティティのための前記ネームスペース登録プロセスの実現を選択的に遅延させることをさらに含む、請求項 9 に記載の 1 つ以上の有形な記憶媒体。

【請求項 11】

以下の複数の動作を実行することによってソフトウェアコンポーネント・エコシステムを実現するための方法であって、前記以下の複数の動作は、

ユーザ入力デバイスから 1 つ以上の信号を受取って、デベロッパエンティティの識別名を規定することと、

20

ユーザ入力デバイスから 1 つ以上の信号を受取って、安全な検証メカニズムを規定することと、

前記安全な検証メカニズムと関連付けてデベロッパエンティティ名を含むカタログエントリを記録することと、

前記カタログエントリと関連付けてトランザクションを実行してソフトウェアコンポーネントを転送することと、

前記カタログエントリを用いて前記トランザクションを検証することと、を含む、方法。

【請求項 12】

前記安全な検証メカニズムは、公開鍵およびデジタル署名を含む、請求項 11 に記載の 1 つ以上の有形な記憶媒体。

30

【請求項 13】

前記ソフトウェアコンポーネント・エコシステムにおける複数のノードに対して前記カタログエントリを伝搬することをさらに含む、請求項 12 に記載の 1 つ以上の有形な記憶媒体。

【請求項 14】

前記複数のノードの各々は、

分散型ブロックチェーン台帳のレプリカまたは近似レプリカを維持する複数のノードと協調して前記分散型ブロックチェーン台帳を実現するためのコードを含むカタログインスタンスを表している、請求項 13 に記載の 1 つ以上の有形な記憶媒体。

40

【請求項 15】

前記エントリにネームスペースを割付けることをさらに含む、請求項 12 に記載の 1 つ以上の有形な記憶媒体。

【請求項 16】

複数の相互通信カタログインスタンスを含む、請求項 1 に記載の分散型ピア・ツー・ピア・コンポーネント・エコシステムのコンポーネントカタログのインスタンスを用いて、

前記デベロッパエンティティ名に関するデベロッパエンティティ登録情報を受取ることと、

前記デベロッパエンティティ登録情報を検証して、前記デベロッパエンティティ登録情報を、デベロッパエンティティ登録の記録またはブロックとして、前記複数の相互通信

50

カタログインスタンスの各々にアクセス可能であるブロックチェーンとして実現される分散型データベースに対してコミットすることにより、前記分散型ピア・ツー・ピア・コンポーネント・エコシステムのフルノードすべてに対して前記デベロッパエンティティ登録情報を伝搬することを実行する、請求項 1 1 に記載の 1 つ以上の有形な記憶媒体。

【請求項 1 7】

前記受取るとは、

前記デベロッパエンティティ名と、登録が実行されているカタログインスタンスに関連付けられた公開鍵と、前記デベロッパエンティティ名に関連付けられたデジタル署名とをデベロッパエンティティにより供給することの後に付なわれ、前記デジタル署名は、前記複数の相互通信カタログインスタンスのカタログインスタンスで、デベロッパエンティティアイデンティティを確認するのに使用可能である、請求項 1 6 に記載の 1 つ以上の有形な記憶媒体。

10

【請求項 1 8】

前記分散型ピア・ツー・ピア・コンポーネント・エコシステムの前記フルノードは、前記複数の相互通信カタログインスタンスを含む、請求項 1 6 に記載の 1 つ以上の有形な記憶媒体。

【請求項 1 9】

ネームスペース登録プロセス中の 1 つ以上の供給されたネームスペースをデベロッパ登録情報に関連付けることによって、

前記デベロッパエンティティ登録情報を伝搬した後、ネームスペース登録プロセスの開始および実現に回答して、1 つ以上のネームスペースをデベロッパエンティティに割付けることをさらに含む、請求項 1 6 に記載の 1 つ以上の有形な記憶媒体。

20

【請求項 2 0】

装置であって、

1 つ以上のプロセッサと、

論理とを含み、前記論理は、前記 1 つ以上のプロセッサによって実行すべき 1 つ以上の有形な媒体において符号化され、前記論理は、実行されると、

ユーザ入力デバイスから 1 つ以上の信号を受取って、デベロッパエンティティの識別名を規定し、

ユーザ入力デバイスから 1 つ以上の信号を受取って、安全な検証メカニズムを規定し

30

、  
前記安全な検証メカニズムと関連付けてデベロッパエンティティ名を含むカタログエントリを記録し、

前記カタログエントリと関連付けてトランザクションを実行して、ソフトウェアコンポーネントを転送し、

前記カタログエントリを用いて前記トランザクションを検証する、

ように動作可能である、装置。

【請求項 2 1】

装置であって、

1 つ以上のプロセッサと、

論理とを含み、前記論理は、前記 1 つ以上のプロセッサによって実行すべき 1 つ以上の有形な媒体において符号化され、前記論理は、実行されると、

前記ソフトウェアコンポーネント・エコシステムの第 1 のノードにおいてトランザクション情報を受取り、

前記第 1 のノードを用いて、前記ソフトウェアコンポーネント・エコシステムを特徴付ける 1 つ以上のイベント、発生、特性または関係の文書化を含む前記第 1 のトランザクションに前記トランザクション情報をパッケージし、

前記第 1 のノードと、前記第 1 のノードと相互に通信する 1 つ以上の追加のノードとによって維持される 1 つ以上の分散型未検証待ち行列に前記トランザクションを選択的に伝搬し、

40

50

前記 1 つ以上の分散型未検証待ち行列において前記第 1 のトランザクションを含む 1 つ以上のトランザクションが、処理されるべきであるとともに、前記第 1 のノードおよび前記 1 つ以上の追加のノードによって維持される分散型ブロックチェーンの新しいブロックに含まれべきであると判断し、

前記 1 つ以上のトランザクションを用いて第 1 のハッシュを演算し、

前記ブロックチェーンの末端ブロックに関連付けられた前のブロックハッシュと前記第 1 のハッシュとを用いて、前記ブロックチェーンの前記新しいブロックについての新しいハッシュを演算し、

前記第 1 のトランザクションを含む前記 1 つ以上のトランザクションを、前記ブロックチェーンのうち前記新しいハッシュを組込んだ前記新しいブロックに組み込み、

10

前記ブロックチェーンの後ろに前記新しいブロックを追加する、

ように動作可能である、装置。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

本願は、2017年6月30日付けで提出された、「分散型台帳を用いて公共のソフトウェアコンポーネント・エコシステムを管理するためのシステムおよび方法 (SYSTEM AND METHOD FOR MANAGING A PUBLIC SOFTWARE COMPONENT ECOSYSTEM USING A DISTRIBUTED LEDGER)」と題された米国仮特許出願連続番号第62/527,479号(弁理士事件番号:ORACP0200P1/OR A180001-US-PSP-1)と、2017年9月1日付けで提出された、「ブロックチェーン特徴を用いて公共のソフトウェアコンポーネント・エコシステムを管理するためのシステム (SYSTEM FOR MANAGING A PUBLIC SOFTWARE COMPONENT ECOSYSTEM USING BLOCKCHAIN FEATURES)」と題された米国仮特許出願連続番号第62/553,406号(弁理士事件番号:ORACP0200P2/OR A180001-US-PSP-2)と、2017年12月1日付けで提出された、「分散型台帳を用いて公共のソフトウェアコンポーネント・エコシステムを管理するためのシステムおよび方法 (SYSTEM AND METHOD FOR MANAGING A PUBLIC SOFTWARE COMPONENT ECOSYSTEM USING A DISTRIBUTED LEDGER)」と題された米国特許出願連続番号第15/829,684号(弁理士事件番号:ORACP0200/OR A180001-USNP)との利益を主張するものであって、あらゆる目的で本願中に十分に記載されるように、引用によってここに援用されている。

20

30

【背景技術】

【0002】

背景

本願は、コンピューティングに関し、より特定的には、エコシステムのパーティシパント間でソフトウェアコンポーネントなどのコンピューティングリソースを共有するのに適したコンピューティング・エコシステムに関する。

【0003】

ソフトウェアコンポーネント・エコシステムを実現および制御するためのシステムおよび方法は、アプリストアおよびオープンソースソフトウェア開発プラットフォームを含む、要求の厳しいさまざまなアプリケーションにおいて採用されている。このようなアプリケーションは、エコシステムにおいて発生するトランザクションを選択的に処理してその明細を明らかにするとともに、トランザクション情報を利用してエコシステムの効率的な動作を確実にするための効率的なメカニズムを必要とすることが多い。

40

【発明の概要】

【発明が解決しようとする課題】

【0004】

従来、コンテンツ(たとえば、ソフトウェアプログラムおよびコンポーネント、データ、オブジェクトなど)をやり取りするためのコンピューティング・エコシステムは、アプ

50

リストアの所有者などによって中心的に制御されているか、または、相対的に制御されていない。しかしながら、中心的に制御されたエコシステムは、相応にロバスタな中央管理インフラストラクチャを実現したり、このようなシステム上のスケーラビリティおよびディザスタリカバリを管理したりするコストや、それらの管理に関する要員配置の全体的コストなどのさまざまな不利益を被る可能性がある。したがって、コンピューティング・エコシステムの効率的な実現および制御を容易にするためのシステムおよび方法が依然として必要とされている。

【課題を解決するための手段】

【0005】

要約

例示的な実施形態は、部分的に分散型ブロックチェーン台帳を用いて、分散型エコシステムにおいて行われるトランザクション、たとえば（以下においてより十分に説明される）さまざまなタイプの登録トランザクションを維持、保証、検証、更新、監視、または他の場合には制御もしくは実施することによって、分散制御を用いてソフトウェアコンポーネント・エコシステムを実現することを容易にするシステムおよび方法を開示する。分散型エコシステムの（この明細書中においてカタログインスタンスと称されるかまたは表現される）エコシステムノードは、1つ以上のトランザクションを含む新しいブロックの追加を含む、共有されたブロックチェーン台帳（この明細書中においては単にブロックチェーンとも称される）を管理するためのソフトウェアを実行する。

【0006】

ソフトウェアコンポーネント・エコシステムの実現を容易にするための例示的な方法は、ユーザ入力デバイスから1つ以上の信号を受取って、デベロッパエンティティの識別名を規定するステップと、ユーザ入力デバイスから1つ以上の信号を受取って、デベロッパエンティティが識別名を登録することが認可されていることを検証するのに使用可能な安全な検証メカニズムを規定するステップと、カタログ（たとえば、ノードおよび付随する分散型台帳を実現するカタログ）に、検証メカニズムに関連したデベロッパ名を含むエントリを記録するステップと、カタログエントリに関連したコンポーネントアップロード、コンポーネントダウンロード、または他のコンポーネント転送を実行して、ソフトウェアコンポーネントを転送するステップと、カタログエントリ（たとえば、デベロッパエンティティ登録トランザクションエントリ）を用いて、コンポーネントアップロード、コンポーネントダウンロードまたは他のソフトウェアコンポーネント転送に関するトランザクションを検証する（たとえば、分散型台帳にトランザクションをコミットする）ステップとを含む。これにより、トランザクションを含むデータが分散型台帳にコミットされる。

【0007】

より具体的な実施形態においては、例示的な方法はさらに、安全な検証メカニズムがデベロッパエンティティに関連付けられた公開鍵およびデジタル署名を含むことを規定する。記録されたカタログエントリが初めに未検証の待ち行列に入れられてから、カタログエントリの複製がソフトウェアコンポーネント・エコシステムにおける複数のノードに伝搬される。複数のノードの各々は、分散型ブロックチェーン台帳のレプリカまたは近似レプリカを維持する他の複数のノードと協調して分散型ブロックチェーン台帳を実現するためのコードを含むカタログインスタンスを表している。

【0008】

したがって、この明細書中に記載される特定の実施形態は、分散型台帳（たとえば、分散型ブロックチェーン）におけるトランザクションデータの登録を容易にするための効率的なメカニズムを提供する。トランザクションに関する情報を用いて、パーティシパントおよび関連するコンポーネントの信頼を確実にすること、コンポーネントのアイデンティティ、保全性、由来または起源、サポート状態などを管理および検証すること、ソフトウェアコンポーネントへの適切なライセンス付与を確実にすること、利用可能なソフトウェアサポートサービスに関する情報を提供すること、コンポーネントのコンシューマおよび/またはコントリビュータなどのユーザを承認すること、矛盾する名前または混乱させる

10

20

30

40

50

名前を持ったコンポーネントが提供されたときに起こる可能性のある問題を防止または軽減すること、ソフトウェアコンポーネントの更新および/またはリコールを容易にすることなどを可能にするが、これらすべての場合、別個に管理されたコアコントローラを用いる必要はない。

【0009】

明細書の残りの部分および添付の図面を参照することによって、この明細書中に開示されている特定の実施形態の性質および利点がさらに理解され得る。

【図面の簡単な説明】

【0010】

【図1】ソフトウェアコンポーネントを選択的に共有するとともに、分散型ブロックチェーン台帳を用いてトランザクションを登録するための、一般化されたピア・ツー・ピア・コンピューティング・エコシステムの第1の例を示す図である。

10

【図2】図1のピア・ツー・ピア・コンピューティング・エコシステムの第1の例の実現例を示すより詳細な図である。

【図3】図1および図2のピア・ツー・ピア・コンピューティング・エコシステムのピアまたはノードとして機能する個々のデベロッパコンピューティングシステムの詳細な例を示す図であって、例示的なユーザインターフェイス（User Interface：UI）制御およびオプションと、ユーザデベロッパコンピューティングシステムがピア・ツー・ピア・エコシステムのノードとして機能するようにユーザデベロッパコンピューティングシステムをピア・ツー・ピア・エコシステムに連結するのに用いられるとともにカタログインスタンスに含まれ得る例示的なモジュールとを含む。

20

【図4】図1～図3の実施形態を用いて分散型ブロックチェーン台帳にデベロッパエンティティを登録するための第2の例示的なエコシステムおよび付随するカタログインスタンスプロセスを示す図であって、第2の例示的なシステムは図1～図3の実施形態に従って実現され得る。

【図5】図4の第2の例示的なエコシステムによって実現可能な第1の例示的な方法を示すフロー図である。

【図6】ソフトウェアコンポーネントのカタログをブラウズするための、かつ、付随するコンピューティングシステムがアクセスできるカタログインスタンスによって維持されるカタログからソフトウェアコンポーネントをサーチ、ダウンロード、編集および削除するためのさまざまなUI制御およびオプションを提供するとともに、図3のユーザデベロッパコンピューティングシステムなどのコンピューティングシステム上に表示され得る例示的なUIディスプレイスクリーンを示す図である。

30

【図7】ピア・ツー・ピア・ソフトウェアコンポーネント・エコシステムの、カタログインスタンスと称されるノードによって維持されるブロックチェーンの初期状態を示す、第3の例示的なエコシステムおよび付随するコンピューティング環境を例示する第1のブロック図である。

【図8】第1の例示的なトランザクション（すなわち、デベロッパ登録トランザクション）が発端元カタログインスタンス（たとえば、第1のカタログインスタンス）によってカタログインスタンスのうち未検証の待ち行列に伝搬された後における、図7の第3の例示的なシステムを示す第2のブロック図である。

40

【図9】デベロッパ登録トランザクションを処理（たとえば、検証または確認）するために（選択ルールまたはアルゴリズムに従って）選択されたカタログインスタンス（たとえば、第2のカタログインスタンス）による、最初のデベロッパ登録トランザクションの検証およびブロックチェーンに対するコミットの処理中における、図8の第3の例示的なシステムを示す第3のブロック図である。

【図10】選択されたカタログインスタンスによるブロックチェーンのブロックの演算中における、図9の第3の例示的なシステムを示す第4のブロック図であって、ブロックは、未検証の待ち行列に、図9のデベロッパ登録トランザクションを含むペンディングトランザクションを含む。

50

【図 1 1】第 2 のカタログインスタンスによって処理されたブロックがブロックチェーンにコミットされて、エコシステムのカatalogインスタンスの各々によって維持されるブロックチェーンレプリカに伝搬された後における、図 1 0 の例示的なシステムを示す第 5 のブロック図である。

【図 1 2】第 2 のカタログインスタンスが第 2 のトランザクション（すなわち、第 1 のネームスペース登録トランザクション）を開始して、ソフトウェアコンポーネント・エコシステムにおける他のカタログインスタンスのうち未検証の待ち行列に第 2 のトランザクションを伝搬した後における、図 1 1 の例示的なシステムを示す第 6 のブロック図である。

【図 1 3】第 3 のカタログインスタンスが第 3 のトランザクション（すなわち、第 2 のネームスペース登録トランザクション）を開始して、これがソフトウェアコンポーネント・エコシステムにおける他のカタログインスタンスのうち未検証の待ち行列に伝搬された後における、図 1 2 の例示的なシステムを示す第 7 のブロック図である。

【図 1 4】ブロックチェーンに含めるべき付加的なブロックを検証するために第 1 のカタログインスタンスが選択された後における、図 1 3 の例示的なシステムを示す第 8 のブロック図であって、付加的なブロックは図 1 2 の第 2 のトランザクションおよび図 1 3 の第 3 のトランザクションを組込むためのものである。

【図 1 5】ソフトウェアコンポーネント・エコシステムのブロックチェーンに追加されるべき付加的なブロックの演算中における、第 1 のカタログインスタンスを示す図 1 4 の例示的なシステムを示す第 9 のブロック図である。

【図 1 6】第 1 のカタログインスタンスが付加的なブロックの処理を完了して、ソフトウェアコンポーネント・エコシステムのブロックチェーンに付加的なブロックを伝搬した後における、図 1 5 の例示的なシステムを示す第 1 0 のブロック図である。

【図 1 7】図 1 ~ 図 1 6 の実施形態によって実現可能な第 2 の例示的な方法を示すフロー図である。

【図 1 8】図 1 ~ 図 1 7 の実施形態を実現するのに使用可能であるシステムおよび付随のコンピューティング環境を示す概略ブロック図である。

【図 1 9】図 1 ~ 図 1 8 の実施形態を実現するのに使用可能なコンピューティングデバイスを示す概略ブロック図である。

【発明を実施するための形態】

【0011】

#### 実施形態の詳細な説明

説明を目的として、ソフトウェアコンポーネント・エコシステムは、分散型コンピューティングリソース（たとえば、カタログインスタンス、付随する分散型ブロックチェーンなど）への / からのソフトウェアコンポーネントのアップロードおよび / またはダウンロードを可能にするように構成された、ネットワーク化された分散型コンピューティングリソースの集合を含むものであれば如何なるコンピューティング環境であってもよい。ネットワーク化されたコンピューティング環境は、相互通信コンピュータ、すなわち、コンピュータネットワーク、たとえば、ローカルエリアネットワーク（local area network : LAN）、ワイドエリアネットワーク（wide area network : WAN（たとえば、インターネット））、クラウドインフラストラクチャおよびサービスなど、を含むものであれば如何なるコンピューティング環境であってもよい。同様に、ネットワーク化されたソフトウェアアプリケーションは、ネットワークを介して 1 つ以上のコンピューティングリソース（たとえば、サーバ）と通信することまたはそれらを用いることを容易にするように適合されたコンピュータコードであってもよい。

【0012】

なお、コンピューティングリソース（たとえば、エコシステムのネットワークを介して相互に通信し得るコンピュータシステム）の集合がこの明細書中においてノードと称されていることに留意されたい。所与のノード（たとえば、ソフトウェアコンポーネントカタログのインスタンス（この明細書中においてはカタログインスタンスと称される））は、他のノードと相互に通信するとともに選択的にデータ（たとえば、エコシステムについて

10

20

30

40

50

の登録情報を含むブロックチェーンのレプリカ)を共有するためのソフトウェアと、(たとえば、さまざまな登録の完了を案内するためのユーザインターフェイスソフトウェアによる)トランザクションの生成を容易にするためのソフトウェアと、エコシステムのルールへの準拠を確実にすることによりピア・ツー・ピア・エコシステムの実現を可能にするためのソフトウェアとを含んでいてもよい。

【0013】

説明の便宜上、ピア・ツー・ピア・ネットワークまたはエコシステムは、コンピュータネットワークにわたって分散されるとともに、プロセスワークロードの共有を容易にするために相互に通信し得るコンピューティングリソース(たとえば、コンピュータシステムおよび/またはソフトウェアアプリケーション)、すなわちノード、の任意の集合であってもよい。

10

【0014】

なお、従来から、ピア・ツー・ピア・ネットワークのピアまたはノードは、ネットワークによって提供されるデータおよび機能にアクセスする同様の特権を有していることに留意されたい。しかしながら、ピア・ツー・ピア・ネットワークのピアまたはノードは、その用語がこの明細書中において用いられている場合、同様に特権が付与される必要はない。たとえば、フルノードと称されるいくつかのノードには最大限に特権が与えられており、すなわち、エコシステムブロックチェーンからの読み出しおよびエコシステムブロックチェーンへの書き込みを行う特権を維持している。さほど特権が与えられていないノードは、エコシステムブロックチェーンにアクセスするためのプロキシとしてフルノードを使用する必要もあるかもしれない。なお、「ピア・ツー・ピア・ネットワーク」および「ピア・ツー・ピア・エコシステム」という語はこの明細書中では同義で用いられる可能性があることに留意されたい。

20

【0015】

説明の便宜上、ソフトウェア機能は、コンピュータコード(すなわち、ソフトウェア)を介して提供される如何なる機能、能力または特徴(たとえば、格納されたデータまたは配置されたデータ)であってもよい。概して、ソフトウェア機能はユーザインターフェイスおよび付随するユーザインターフェイス制御および特徴を用いることによってアクセス可能になり得る。ソフトウェア機能は以下の動作を含み得る。たとえば、コンピューティングオブジェクト(たとえば、トランザクションに関連付けられたビジネスオブジェクト)に関するデータを検索する動作、企業の人員を昇進させたり、採用したり解雇したりするなどの企業関連のタスクを実行する動作、注文する動作、分析を計算する動作、特定のダイアログボックスを開始する動作、検索を実行する動作、などである。

30

【0016】

ブロックチェーンは、ブロックと称される、リンクされた記録の順番付けされたリストであってもよく、この場合、ブロックチェーンは、新しいブロックをブロックチェーンの終端に加えることによって発展させることができるが、より前のブロックの挿入は、より後のブロックが最初にブロックチェーンから展開されないかまたは削除されない限り、禁止される。ブロックチェーンのうちさまざまなブロックには、ブロックチェーンへの組み込み時にタイムスタンプが付されることが多い。ブロックチェーンは、たとえば、所与のコンピューティング環境またはエコシステムのノードにインストールされ得る分散型ソフトウェアアプリケーションまたはネットワーク化されたソフトウェアアプリケーションを用いて実現されてもよい。ブロック間のリンクは、新しいブロックに適用される1つ以上のハッシュを実現することによって実現されてもよい。この場合、1つ以上のハッシュは、1つ以上の前のブロックからの情報を活用または使用する。ブロックチェーンを用いることでトランザクションの分散型台帳を実現することができる。

40

【0017】

説明の便宜上、分散型台帳は、ネットワークの複数のノード間で共有される共有デジタルデータの集合であってもよく、そのコピーが複数のノード間で複製されて格納され得る。分散型台帳によって維持されるデータはノード間で同期されてもよい。

50

## 【 0 0 1 8 】

したがって、分散型台帳は、一種の分散型データベース（すなわち、複数のエンティティのネットワークに連結されたさまざまなエンティティ間でデータを格納するためのメカニズム）として機能し得る。ノードは、ネットワークに連結される如何なるコンピュータシステムおよび/またはソフトウェアアプリケーションおよび/またはソフトウェアシステムまたはこれらのグループであってもよい。ここに記載されるノードは一般に「カタログインスタンス」と称される。というのも、これらノードは、付随するコンピューティング・エコシステムの他のノードおよび/またはパーティシパントがカタログに格納されたデータにアクセスするのを容易にするからである。

## 【 0 0 1 9 】

トランザクションは、エコシステムの1つ以上の局面を記述しているイベント、状態、プロパティまたは他の情報を記載する情報の任意の集合であり得る。この場合、1つ以上の局面は、参加しているデベロッパエンティティ、ソフトウェアコンポーネントコンシューマエンティティ、コントリビュータエンティティ、プロキシ化されたエコシステムパーティシパントおよびシステム、ソフトウェアコンポーネント相互関係、ソフトウェアコンポーネントのダウンロードおよび/またはアップロードのインスタンス、ソフトウェアコンポーネントのサポート状態、コンポーネント由来情報などを含み得る。トランザクションは、この用語が用いられている文脈に応じて、エコシステムにおける活動（たとえば、デベロッパエンティティ登録、ネームスペース登録、コントリビュータ登録など）を記述するデータの集合を指すこともある。または、代替的には、トランザクションは、実際の活動（たとえば、コンポーネントをダウンロードすること）を指すこともある。

## 【 0 0 2 0 】

活動またはタスクを表しているトランザクションは十分に自動化されてもよく、またはマニュアルでの承認または他の検証活動などの人のワークフロータスクを含んでいてもよい。したがって、トランザクションはブロックチェーンにおける単一のもの（たとえば、情報の集合）として表わされる可能性もあるが、トランザクションのいくつかの形態は、実際には、ワークフローが処理されるのに応じて台帳中に記録することができる別個のサブトランザクションに分解されてもよい。したがって、「トランザクション」という語はまた、その語が用いられている文脈に応じて、情報（たとえばコンピューティングオブジェクト）の集合を伝達する動作を指している可能性もあり、情報（たとえばコンピューティングオブジェクト）の実際の集合を指している可能性もある。

## 【 0 0 2 1 】

たとえば、個々のソフトウェアデベロッパ（たとえば、コンポーネントコントリビュータ）がコンポーネントエコシステムに登録する場合、この明細書中に記載されているように、コントリビュータ登録プロセスに関する（たとえば、これを文書化する）情報は、エコシステムのブロックチェーンに組込むための準備として、カタログインスタンスのうち1つ以上の未検証の待ち行列に伝搬されてもよい。情報の収集および/または転送はトランザクションと称されてもよく、収集された情報を維持するコンピューティングオブジェクトはまた、トランザクション（たとえば、デベロッパ登録トランザクション）と称されてもよい。図1を参照すると、このようなトランザクションは、（たとえば、登録機能16によって）収集され得るとともに、分散型台帳14のブロックチェーン20のうち1つ以上のブロックに選択的にコミットされ得る。

## 【 0 0 2 2 】

なお、ブロックチェーン20および分散型台帳14が特定の実現例において同等であり得ることに留意されたい。しかしながら、この例示的な実施形態においては、分散型台帳14はブロックチェーン20を含むものとして示されている。というのも、分散型台帳14は、ブロックチェーン20の単なる実現例とは対照的に、付加的な機能を含み得るからである。たとえば、特定の実現例においては、分散型台帳14は、パーティシパントコンピュータシステム12のノードによってローカルで使用される他のデータベースフォーマットにブロックチェーン20を変換することを可能にするための付加的なコードを含み得

10

20

30

40

50

るとともに、カタログインスタンス・データストア 18 の別々のインスタンスに対してコンポーネントを選択的に分散させることを可能にし得る。

【0023】

所与のノードには、所与のコンピューティング環境またはエコシステムにおいて異なる特権が割付けられてもよい。同様の特権を持つノードは、それが1つ以上の特定のタスクの実現に関係しているため、タスクを完了させるためのピアと称されている。説明の便宜上、ピア・ツー・ピア・エコシステムは、少なくとも部分的に、エコシステムの別々のノードまたはピアを介して実現される1つ以上の分散型またはネットワーク化ソフトウェアアプリケーションによって実現される如何なるエコシステムまたはコンピューティング環境であってもよい。

10

【0024】

この明細書中に記載されるさまざまな例示的な実施形態は、ソフトウェアコンポーネントカタログインスタンスのノードを含むピア・ツー・ピア・ソフトウェアコンポーネント・エコシステムによって実現される。以下により十分に記載される例示的なソフトウェアコンポーネントカタログインスタンスはさまざまなソフトウェアアプリケーションを実行し得る。これらさまざまなソフトウェアアプリケーションは、(ソフトウェアコンポーネントのデータベースを含み得る)ローカルデータストアを維持および管理するためのソフトウェア、セキュリティおよび許可機能を実現するためのソフトウェア、さまざまなタイプの登録(それらの例が以下においてより十分に記載されている)を可能にするためのユーザインターフェイス(UI)ディスプレイスクリーンを生成するためのソフトウェア、エコシステムのための未検証のトランザクション待ち行列を管理するためのソフトウェア、他のカタログインスタンスと通信するためのソフトウェア、エコシステムブロックチェーンのレプリカを維持するためのソフトウェア、エコシステムのブロックチェーンのための新しいブロックをコンピューティングする(すなわち、検証するかまたは確認する)ためのソフトウェア、エコシステムの1つ以上のカタログインスタンスによる検証(およびブロックチェーンブロックへの包含)のためのトランザクションをサブMITTするためのソフトウェア、1つ以上のブロックをブロックチェーンに追加するのに必要な演算を実行するためにカタログインスタンスを選択するための任意のアルゴリズムを実現するためのソフトウェア、ブロックをブロックチェーンに追加するのに必要なハッシュを演算するためのソフトウェア、などを含み得る。

20

30

【0025】

概して、デベロッパおよび/またはビジネスのコミュニティは、ソフトウェアコンポーネント・エコシステムを用いて、情報、リソースおよびコンポーネントのやり取りを可能にするかまたは容易にする共通の技術的プラットフォームを用いてソフトウェアおよびサービスのための共有の市場と協働的に対話してもよい。

【0026】

ソフトウェアコンポーネント・エコシステムは、デベロッパ、ベンダーおよび顧客によって使用される再使用可能なソフトウェアコンポーネントのオープンなエコシステムとして実現され得る。このようなエコシステムは、ネットワーク化されたインフラストラクチャまたは「クラウド」インフラストラクチャならびに付随するプロセスおよびサービスのまわりに構築されてもよい。しかしながら、本発明の特定の実施形態は特定の処理プラットフォーム、技術およびインフラストラクチャに関して記載され得るが、他の変形例も可能であり、異なるシステムに適合されてもよい。

40

【0027】

ソフトウェアコンポーネント・エコシステムについての1設計は、信頼、アイデンティティ、保全性、サポート、ネームスペース、コンポーネントのアイデンティティ、知的財産、製品リコール、由来およびライセンスなどのさまざまな局面の検討事項を含んでいる。他の局面についての特徴を含むこともできる。さまざまな特徴および局面が特定の実施形態に関連付けて記載されているが、特定の実施形態がさまざまな異なるセットの記載特徴を使用できることが明らかになるはずである。特徴は省略または変更されてもよく

50

、付加的な特徴が含まれてもよい。

【0028】

この明細書中に記載される特定の実施形態は、1つ以上の信号をユーザ入力デバイスから受取ってデベロッパエンティティの識別名を規定すること；1つ以上の信号をユーザ入力デバイスから受取って安全な検証メカニズムを規定すること；検証メカニズムに関連したデベロッパ名を含むエントリをカタログに記録すること；カタログエントリに関連したトランザクションを実行してソフトウェアコンポーネントを転送すること；および、カタログエントリを用いてトランザクションを検証すること；によってソフトウェアコンポーネント・エコシステムを実現するための装置、記憶媒体および方法を提供する。付加的な特徴は、カタログにおけるエントリによって規定されるエンティティにネームスペースを割付けることを可能にする。

10

【0029】

従来より、ソフトウェアデベロッパは、ソフトウェアアプリケーションの開発および関連するファイルの格納を容易にするために特定のクラウドサービスをサブスクリブし得る。ソフトウェアアプリケーションまたはプロセスフロー開発のために構成されるクラウドサービスは、ここではプロセスクラウドサービス（Process Cloud Service：PCS）と称される。

【0030】

プロセスクラウドサービスは、開発されている所与のソフトウェアプログラムによって用いられるファイルおよび他のオブジェクトを格納するためにネットワーク化されたデータベースを使用する可能性がある。サーバ側開発環境はブラウザを介してデベロッパにアクセスであってもよい。開発環境は、開発されたソフトウェアアプリケーションファイルがPCSデータベースに格納されるように、PCSにバックアップされていてもよい。

20

【0031】

説明の便宜上、コンピューティング環境は、コンピュータ処理を伴う1つ以上のタスクを実行するために用いられるコンピューティングリソースの如何なる集合であってもよい。コンピュータはメモリと通信する如何なるプロセッサであってもよい。コンピューティングリソースは、コンピューティング環境の如何なるコンポーネント、メカニズムまたは能力もしくはその量であってもよく、プロセッサ、メモリ、ソフトウェアアプリケーション、ユーザ入力デバイスおよび出力デバイス、サーバなどを含むがこれらに限定されない。コンピューティングリソースの例は、1つ以上のウェブサービス、アプリケーションプログラミングインターフェイス（Application Programming Interface：API）などによって提供されるデータおよび/またはソフトウェア機能を含む。

30

【0032】

エンタープライズコンピューティング環境はビジネスまたは組織のために用いられる如何なるコンピューティング環境であってもよい。一例としてのエンタープライズコンピューティング環境は、ネットワークにわたって分散されたさまざまなコンピューティングリソースを含み、さらに、イントラネットウェブサーバ、データベース、ローカルのハードディスクまたはファイルサーバ上のファイル、電子メールシステム、ドキュメント管理システム、ポータルなどの上にあるプライベートの共有コンテンツを含み得る。

40

【0033】

所与のソフトウェアアプリケーションは、構成するソフトウェアアプリケーションまたはモジュール（たとえば、サービス、機能、手順、コンピューティングオブジェクト、など）を含み得る（が、必ずしも含むものではない）。したがって、「ソフトウェアアプリケーション」という語は、ネットワーク化されたソフトウェアアプリケーションまたはそれらが一体化されたグループを含んでいてもよい。

【0034】

この明細書中に記載される特定の実施形態は、プロセスベースのソフトウェアアプリケーションの開発、展開および実現に特に有用である。プロセスベースのソフトウェアアプリケーションは、プロセス要素またはソフトウェア活動とも称されるステップの1つ以上

50

のシーケンスによって定義可能な如何なるソフトウェアアプリケーションであってもよい。「プロセス要素」、「フロー要素」、「ソフトウェア要素」および「ソフトウェアプロセス要素」という語はこの明細書中では同義であって、たとえば、イベント、活動、ゲートウェイ、サブプロセスなどの任意のステップを指していることもある。プロセスベースのソフトウェアアプリケーションのステップのシーケンスはプロセスフローと称されることもある。プロセスフローは、ユーザインターフェイス（UI）ディスプレイスクリーンにおけるスイムレーンを介してモデル化されて例示されることが多い。プロセスベースのアプリケーションは、さまざまなプロセスステップのためのさまざまなウェブサービスおよび関連するソフトウェアコンポーネントを利用し得る複合アプリケーションによって実現されることが多い。

10

**【0035】**

一般に、ソフトウェアおよびプロダクト開発の普及形態は「ソフトウェア・エコシステム」と称され得る。ソフトウェアエコシステムは、ソフトウェアおよびサービスのための共有市場と協働的に機能して対話する1セットのデベロッパおよび/またはビジネスを、それらの間の関係とともに含み得る。これらの関係は、共通の技術的プラットフォームを含み得るとともに、情報、リソースおよびコンポーネントのやり取りによって機能し得る。

**【0036】**

ソフトウェアコンポーネント・エコシステムは特定タイプのソフトウェア・エコシステムであってもよく、主な特徴として、ソフトウェアコンポーネントの開発および交換を含む。ソフトウェアコンポーネント・エコシステムにおいては、デベロッパ、パートナーおよび顧客によって使用されるように再使用可能なソフトウェアコンポーネントのオープンエコシステムを促進することが望ましいかもしれない。しばしば、このようなエコシステムはネットワーク化されたまたは「クラウド」のインフラストラクチャ、プロセスおよびサービスのまわりに構築されることとなる。しかしながら、本発明の特定の実施形態は、特定の処理プラットフォーム、技術およびインフラストラクチャに関連付けて記載され得るが、他の変更も可能であり、異なるシステムに適合されてもよい。

20

**【0037】**

明瞭にするために、いくつかの周知のコンポーネント、たとえば、ハードドライブ、プロセッサ、オペレーティングシステム、電源、ルータ、インターネット・サービス・プロバイダ（Internet Service Provider：ISP）、ワークフロー・オーケストレータ、プロセススケジューラ、プロセスクラウド、ビジネスプロセスマネジメントシステム、エコシステムデベロッパエンティティ、エコシステムコントリビュータエンティティ、エコシステムコンシューマ、統合開発環境、プロキシシステム、アイデンティティ管理システム、認証機関（Certificate Authority：CA）などは、図中に必ずしも明確にコールアウトされているわけではない。しかしながら、本教示を利用できる当業者であれば、所与の実現例についてのニーズを満たすためにいずれのコンポーネントを実装するべきかと、それらコンポーネントを如何に実装するべきかについて認識するだろう。

30

**【0038】**

図1は、（1つ以上のカタログインスタンス・データストア18に格納されるような）ソフトウェアコンポーネントを選択的に共有するための、かつ、分散型ブロックチェーン台帳14および20を用いてトランザクションを登録するための、一般化されたピア・ツー・ピア・コンピューティング・エコシステム10の第1の例を示す。

40

**【0039】**

なお、一般的に、システム10のさまざまなモジュールのグループ分けは例示的なものであって、さまざまに異なってもよく、たとえば、この教示の範囲から逸脱することなく、特定のモジュールが他のモジュールと組み合わせられてもよく、もしくは他のモジュール内に実装されてもよく、または、それ以外の場合には、モジュールが、ネットワーク間で、もしくは1つ以上のコンピューティングデバイスもしくは仮想マシン内において（図示されるのとは）異なる態様で分散されていてもよい。たとえば、例示的な登録機能16

50

、カタログインスタンス・データストア 18 および分散型台帳 14 は、パーティシパントコンピュータシステム 12 とは（明瞭さのために）別個に示されているが、実際には、このようなエンティティは、関連するピア・ツー・ピア・ネットワークに關与している 1 つ以上のパーティシパントコンピュータシステム 12 によって実現されている。

【0040】

パーティシパントであるシステム 12 は、デベロッパエンティティ（たとえば、個々のデベロッパが属する企業または組織）を含み得る。デベロッパエンティティは、コード（たとえばソフトウェアコンポーネント）を 1 つ以上のいわゆるカタログインスタンスに付与する）いわゆるコントリビュータを含み得る。これら 1 つ以上のいわゆるカタログインスタンスを用いることで、登録機能 16 を実現するとともに、デベロッパが 1 つ以上のカタログインスタンス・データストア 18 において維持されているソフトウェアコンポーネントとの対話（たとえば、ダウンロード、アップロードなど）を行うこと可能にする。デベロッパエンティティおよび関連するデベロッパはまた、ローカルのカタログインスタンスからソフトウェアコンポーネントをダウンロード（すなわち、消費）してもよい。

10

【0041】

カタログインスタンス・データストア 18 からソフトウェアコンポーネントをダウンロードする人またはエンティティはコンシューマと呼ばれる。カタログインスタンス・データストア 18 にソフトウェアコンポーネントをアップロードする人またはエンティティは、コントリビュータと呼ばれる。一般に、デベロッパエンティティは、この明細書中においてその語が用いられている場合、コントリビュータおよびコンシューマの両方、またはコントリビュータだけを含む可能性がある。しかしながら、ソフトウェアコンポーネントのコンシューマとしてのみ機能するエンティティはこの明細書中ではデベロッパエンティティとは呼ばれない。

20

【0042】

デベロッパエンティティがたとえば登録機能 16 を用いてカタログインスタンスに登録する場合、および、そのコンシューマがローカルのカタログインスタンス・データストア 18 にソフトウェアコンポーネントをダウンロードする場合、または、そのコントリビュータがローカルのカタログインスタンス・データストア 18 からソフトウェアコンポーネントをアップロードする場合、これらの活動に関するデータは、検証されると、分散型台帳 14 および付随する検証済みブロックチェーン 20 に登録される。

30

【0043】

なお、特定の実施形態においては、デベロッパエンティティの登録がいわゆるデベロッパエンティティ登録トランザクションを生成することに留意されたい。デベロッパエンティティ登録トランザクションの後には、（代替的には、特定の実現例におけるデベロッパエンティティ登録の一部であり得る）ネームスペース登録が続けて行われてもよい。たとえば、登録機能 16 を用いるネームスペース登録の間、ネームスペースはデベロッパエンティティに關連付けられ、対応する情報はネームスペース登録として分散型台帳 14 に格納される。

【0044】

デベロッパエンティティ登録および関連するネームスペース登録の後、個々のコントリビュータおよびコンシューマは、さらに、関連するデベロッパエンティティおよびネームスペースに關連付けて登録してもよい。コントリビュータが、既に登録済みのソフトウェアコンポーネントを別のカタログインスタンス（すなわち、デベロッパエンティティが直接通信しているローカルのカタログインスタンス以外のカタログインスタンス）にアップロードすると、別個のアップロードトランザクションが使用可能となる。

40

【0045】

コンポーネントのダウンロードトランザクション、コンポーネントの使用登録などを含むがこれらに限定されない他のさまざまなタイプの登録も可能である。以下においてより十分に説明されるように、一般に、各々のタイプの登録に関する情報は、検証後、分散型台帳 14 のブロックチェーン 20 にコミットされる。

50

## 【 0 0 4 6 】

分散型台帳 1 4 は、管理上（たとえば、ノード管理、アイデンティティ管理など）のタスクを（たとえば、図 1 においてアイテム 1 4 ~ 1 8 によって集合的に表されている）カタログインスタンスおよび / またはローカルのアイデンティティドメインに対して委譲することを可能にするとともに、パーティシパントが、すなわち分散型台帳 1 4 のブロックチェーン 2 0 に登録されたトランザクションによって、関連するエコシステム 1 0 に関するトランザクションについての整合性のあるビューまたは概要にアクセスすることも可能にする。

## 【 0 0 4 7 】

この例示的な実施形態においては、分散型台帳 1 4 内のトランザクションは、エンティティおよび / またはそれにサブミットされたトランザクションのレジストラに関連付けられた役割に従って許可される。トランザクションは、トランザクションに用いられるノード（たとえばカタログインスタンス）の性質に従ってさらに許可されてもよい。たとえば、いくつかのノードがフルノードであり得るのに対して、他のノードは、ブロックチェーン 2 0 のブロックを検証する際にアクセスして関与するための特権が削減されてしまっている可能性もある。

## 【 0 0 4 8 】

異なる態様で許可され得る例示的な役割は、（デベロッパエンティティアイデンティティの検証、デベロッパエンティティに対するネームスペース割付けの検証などを担う可能性のある）コミュニティリーダーの役割；（デベロッパエンティティのアイデンティティ、コントリビュータ、コンポーネントのアイデンティティなどの登録および / または管理を担う可能性のある）デベロッパの役割；（コンポーネント使用登録を担う可能性のある）ディストリビュータの役割；（コンポーネント使用登録に関与し得る）コンシューマ、などを含む。

## 【 0 0 4 9 】

なお、エコシステム 1 0 などの分散型システムおよびピア・ツー・ピアシステムにおいては、トランザクション登録および関連するデータ登録が重要となり得ることに留意されたい。データ（たとえば、登録および関連するソフトウェアコンポーネントに関するデータ）は複数の位置に存在している。すなわち、データは、パーティシパントコンピュータシステム 1 2 のフルノード間で複製される。

## 【 0 0 5 0 】

したがって、データ（たとえばトランザクションデータ）が 1 つのノードによって変更される場合、この変更（および関連するトランザクション）がパーティシパントコンピュータシステム 1 2 のフルノードすべてに伝搬されることが重要であり得る。トランザクションデータがソフトウェアコンポーネント登録に関係している場合、分散型台帳 1 4 および付随するブロックチェーン 2 0 にコミットされるトランザクションに実際のソフトウェアコンポーネント全体を格納することは非実用的であるかもしれない。したがって、所与のソフトウェアコンポーネントを格納するために用いられるファイル（すなわち分散媒体）のハッシュは、ブロックチェーン 2 0 に、すなわち、ソフトウェアコンポーネント登録のために、登録される可能性がある。

## 【 0 0 5 1 】

図 2 は、図 1 のピア・ツー・ピア・コンピューティング・エコシステム 1 0 の第 1 の例の実現例 3 0 を示すより詳細な図である。図 1 および図 2 を参照すると、図 2 の実現例 3 0 は、デベロッパエンティティシステム 2 2 と、図 1 のパーティシパントコンピュータシステム 1 2 を含む外部のプロキシ化されたシステム 5 0 とを含むものとして示されている。

## 【 0 0 5 2 】

デベロッパエンティティシステム 2 2 は、分散型台帳 1 4 を実現するパーティシパントシステム（たとえば、図 1 のシステム 1 2 ）のクラウド 5 4 と通信する。この実現例 3 0 においては、クラウド 5 4 はさらに、1 つ以上のアイデンティティドメイン 4 4 を含む。

10

20

30

40

50

1つ以上のアイデンティティドメイン44は、コミットされたトランザクションを読取る目的のために、かつ、ブロックチェーン20にトランザクションをコミットするために、(たとえばカタログインスタンス32を介して)分散型台帳14および付随するブロックチェーン20へのアクセスを可能にするよりも前に、デベロッパエンティティおよび個々のユーザを認可するために用いられてもよい。

【0053】

なお、特定の実現例において、(他のカタログインスタンスと選択的に対話するために、かつ、予め定められた許可に関連付けられた予め定められたセットの機能を選択的に実現するために)カタログインスタンス32も認証および許可され得ることに留意されたい。

10

【0054】

例示的なデベロッパエンティティシステム22は、デベロッパエンティティ(たとえば、付随のピア・ツー・ピア・ネットワークおよび関連するエコシステム30のさまざまな動作に参与することができる企業または組織)の1つ以上のコンピューティングシステムを表している。説明の便宜上、デベロッパエンティティは、カタログインスタンス32の1つ以上を用いてこのようなエンティティとして登録することができる、1以上の人々のグループならびに付随するコンピューティングシステムおよびコンピューティングリソースを表しているもよい。この例示的な実施形態においては、デベロッパエンティティは、たとえば、デベロッパエンティティシステム22のデベロッパエンティティと関連付けて登録される個々のコンピュータおよび/またはコンシューマによって、カタログインスタンス32のローカルコンポーネントカタログ36からソフトウェアコンポーネントをアップロード(付与)するとともにソフトウェアコンポーネントをダウンロード(消費)するための特権に関連付けられている。

20

【0055】

デベロッパエンティティ(たとえば、企業)は、たとえば企業の名前を含み得るネームスペースに関連付けられている。所与のネームスペースは名前の階層構造を含み得る。名前の階層構造は、(所与のデベロッパエンティティの階層的ネームスペースにおける最高レベルノードであり得る)デベロッパエンティティの名前に関連付けられたさまざまなソフトウェアコンポーネントの名前を含み得る。

【0056】

なお、デベロッパエンティティおよび関連するシステム22だけでなく、他のタイプのエンティティも、この教示の範囲から逸脱することなくエコシステム30に参与し得ることに留意されたい。なお、デベロッパエンティティシステム22および(1つ以上のカタログインスタンスを用いるためにデベロッパエンティティシステム22のうち1つ以上によってプロキシされている)外部のプロキシ化されたシステム50だけが図2に示されているが、他のタイプのシステムもエコシステム30に参与し得ることに留意されたい。

30

【0057】

この例示的な実施形態においては、デベロッパエンティティシステム22は、エコシステム30のフルノードをホストする1つ以上のコンピューティングシステム(たとえば、実際のコンピュータおよび/または仮想マシン)を含む。フルノードはカタログインスタンス32によって表わされている。説明の便宜上、フルノードは、データ(たとえば、トランザクションデータ)にアクセスするとともに当該データをブロックチェーン分散型台帳14に提供するための特権の十分な読取り、書込みおよびコミットを行うものであれば如何なるノードであってもよい。

40

【0058】

例示的なカタログインスタンス32は、付随するエコシステム30の他のノード(カタログインスタンス)に複製されるかまたは伝搬される、ブロックチェーン分散型台帳14のレプリカである台帳レプリカ34を含む。台帳レプリカは未検証の待ち行列38を含む。未検証の待ち行列38は、ブロックチェーンレプリカ40へのコミットのためにペンディング中である1つ以上の未検証のトランザクションを含み得る。なお、ローカルの未検

50

証の待ち行列 3 8 およびブロックチェーンレプリカ 4 0 のデータが他のノード間で共有されるかまたは他のノードに伝搬され、その後、クラウド 5 4 のうちの分散型の未検証の待ち行列 4 6 および関連するブロックチェーン分散型台帳ブロックチェーン 2 0 を表すことになることに留意されたい。

【 0 0 5 9 】

例示的なカタログインスタンス 3 2 はさらに、ローカルコンポーネントカタログ 3 6 を含むかまたはローカルコンポーネントカタログ 3 6 と通信する。ローカルコンポーネントカタログ 3 6 は、カタログインスタンス 3 2 にアップロードされたソフトウェアコンポーネントを維持する。このソフトウェアコンポーネントは、ローカルコンポーネントカタログ 3 6 におけるコンポーネントの特権コンシューマによってダウンロードされる可能性もある。以下においてより十分に説明されるように、ローカルコンポーネントカタログ 3 6 との各々の対話には、対話を文書化する関連トランザクションが付随している。

10

【 0 0 6 0 】

所与のトランザクションは、最初に、未検証の待ち行列 3 8 にロードされ、他のカタログインスタンス ( ノード ) の他の未検証の待ち行列 4 6 に伝搬され、次いで、トランザクション妥当性チェックが実行されると、代表的な分散型ブロックチェーン 2 0 全体にコミットされる。トランザクション・コミットプロセスは、トランザクションの集合についての 1 つ以上のハッシュと、トランザクションの集合がコミットされることとなるブロックチェーン 2 0 の関連ブロックについての 1 つ以上のハッシュとを計算するために、1 つ以上のカタログインスタンス 3 2 を用いることを必要とする。ブロックチェーン 4 0 および 2 0 にコミットされるべきトランザクションの集合は、未検証の待ち行列 3 8 および 4 6 から選別される。

20

【 0 0 6 1 】

デベロッパエンティティシステム 2 2 はさらに、1 つ以上のカタログインスタンス 3 2 と通信する 1 つ以上のレジストラコンピューティングシステム 4 2 を含む。この例示的な実施形態においては、レジストラシステム 4 2 は、1 つ以上のカタログインスタンス 3 2 に登録をサブミットするために、認可されている ( かまたは認可された人によって用いられている ) 任意のコンピューティングシステムであり得る。なお、デベロッパエンティティシステム 2 2 のうちの或るデベロッパエンティティの 1 つ以上のコンピューティングシステム上で実行されるカタログインスタンス 3 2 がデベロッパエンティティシステム 2 2 のローカルデベロッパエンティティである限りにおいては、ローカルカタログインスタンスと称されることに留意されたい。

30

【 0 0 6 2 】

なお、レジストラシステム 4 2 のレジストラが 1 つ以上のクラウドベースのアイデンティティドメイン 4 4 を介するデベロッパエンティティシステム 2 2 のデベロッパエンティティの代わりに、レジストラとして機能することが認証され得ることに留意されたい。レジストラは、デベロッパエンティティ機密 ( たとえば、公開鍵インフラストラクチャ ( public key infrastructure : P K I ) を用いる場合、秘密鍵 ) を維持し得る。

【 0 0 6 3 】

レジストラシステム 4 2 は、デベロッパエンティティシステム 2 2 のデベロッパエンティティの一部である、個々のデベロッパシステム 2 4 ( たとえば、個々のソフトウェアコンポーネントコントリビュータおよび / またはコンシューマによって用いられるコンピューティングシステム )、他のコントリビュータシステム 2 6 ( たとえば、プロキシ化されたアップローダ )、および、他のユーザーシステム 2 8 ( たとえば、プロキシ化されたダウンロード ) のための ( たとえば、トランザクション登録のための ) プロキシとして機能し得る。

40

【 0 0 6 4 】

さらに、特定の実現例においては、レジストラシステム 4 2 はまた、必ずしもデベロッパエンティティシステム 2 2 の一部ではない外部のプロキシ化されたシステム 5 0 のための 1 つのプロキシ ( または複数のプロキシ ) として機能してもよい。たとえば、外部の統

50

合開発環境 (Integrated Development Environment : IDE) ) は、場合によっては、カタログインスタンス 3 2 および付随するローカルコンポーネントカタログ 3 6 のうち 1 つ以上と対話するインターフェイスとして、デベロッパエンティティのレジストラシステム 4 2 の使用が認可されてもよい。

【 0 0 6 5 】

例示的な使用事例のシナリオにおいては、図 4 および図 5 に関連付けて以下により十分に記載されるように、デベロッパエンティティ (たとえば、デベロッパエンティティシステム 2 2 の所有者) は、初めに、デベロッパエンティティ登録と称されるプロセスにおいてデベロッパエンティティとしてカタログインスタンス 3 2 に登録する。次いで、関連するデベロッパエンティティ登録情報はトランザクションとしてパッケージされるとともに、トランザクションに対する妥当性チェックと、ローカルのブロックチェーン 4 0 および付随する分散型ブロックチェーン 2 0 に対する検証済みトランザクションとしての最終的なコミットとに備えて、ローカルのブロックチェーン台帳レプリカ 3 4 の未検証の待ち行列 3 8 にサブミットされる。

10

【 0 0 6 6 】

デベロッパエンティティ登録の後には、さまざまな追加のタイプの登録 (たとえば、デベロッパエンティティネームスペース登録) が続けられてもよい。これにより、デベロッパエンティティシステム 2 2 のデベロッパエンティティの名前 (および / または他のネームスペース) が登録される。

20

【 0 0 6 7 】

個々のコントリビュータ (たとえば、デベロッパエンティティシステム 2 2 のデベロッパエンティティの個々のデベロッパ) についての登録が次に行なわれてもよい。コントリビュータは、登録済みのデベロッパエンティティ名および / または追加の (コントリビュータによって) 要求されたネームスペースに関連付けて、ブロックチェーン 2 0 に登録される。特定のコントリビュータについての (たとえば、アイデンティティドメイン 4 4 および付随の認証機関 (CA) によって獲得され得るような) 公開鍵およびデジタル署名を提供することによっても、登録の認証を確認すること、すなわち、コントリビュータが認可されて、カタログインスタンス 3 2 とローカルのソフトウェアコンポーネントカタログ 3 6 によって与えられる付随するソフトウェアコンポーネント機能 (たとえば、ソフトウェアコンポーネントのダウンロードおよびアップロード) との使用のために適切に許可されることを確認すること、が容易になり得る。

30

【 0 0 6 8 】

他のタイプの登録および関連するトランザクションは、(たとえば、ローカルコンポーネントカタログ 3 6 にソフトウェアコンポーネントをアップロードする準備のために行われ得るような) ソフトウェアコンポーネント登録と、オプションのコンポーネント使用登録と組合わせたコンシューマコンポーネントダウンロード登録と、コンポーネント使用登録と、を含む。

【 0 0 6 9 】

他のタイプの登録も、依然として、この教示の範囲から逸脱するなく実現可能である。たとえば、(たとえば、1 つのソフトウェアコンポーネントが別のソフトウェアコンポーネント内で用いられる場合の) ソフトウェアコンポーネントのネスト型使用を追跡する登録、ローカルコンポーネントカタログ状態情報の登録 (たとえば、ローカルコンポーネントカタログ 3 6 における利用可能なソフトウェアコンポーネントのリスト化)、与えられた機能、などである。

40

【 0 0 7 0 】

一般に、ブロックチェーン分散型台帳 1 4 にさまざまなタイプの登録情報を収集することにより、エコシステム 3 0 全体のために多くの利点が見られる可能性がある。たとえば、この明細書中に記載される実施形態を用いることにより、ソフトウェアコンポーネント間の名前の不一致を回避すること、特定のコンポーネントが第三者サプライヤによってサポートされることを、特定のコンポーネントが如何にサポートされるかと特定のコンポー

50

ネットを如何に選択的にリコールするかとに鑑みて確実にすることを含む、コンポーネントサポートサービスの効率的な管理、コンポーネントの不正変更を検出すること、知的所有権を実施してコンポーネントの由来および所有権を追跡すること、特定のコンポーネントのためのソフトウェアライセンスが有効であるどうかをチェックすること、コンポーネント有効性の検証を全体的に可能化すること等、が容易になり得る。これはすべて、この明細書中に記載されるように、分散型カタログインスタンス32を用いることによって実質的に置換および改善され得る従来の中心制御メカニズムを必要とすることなく、達成することができる。

#### 【0071】

図3は、図1および図2のピア・ツー・ピア・コンピューティング・エコシステム10および30のピアまたはノードとして機能する個々の例示的なデベロッパコンピューティングシステム62の詳細例を示しており、例示的なUIディスプレイスクリーン70の例示的なUI制御およびオプションと、カタログインスタンス32に含まれ得る例示的なモジュール34~40および60~68とを含んでいる。カタログインスタンス32を用いることで、(デベロッパシステム62と組み合わせられてクラウド14によって表され得る)ピア・ツー・ピア・エコシステムに個々のデベロッパコンピューティングシステム62を連結し、これにより、個々のデベロッパコンピューティングシステム62および付随するカタログインスタンス32がピア・ツー・ピア・エコシステム全体のノードとして機能する。

#### 【0072】

なお、例示のために、個々のデベロッパシステム62上で実行されているカタログインスタンス32が示されることに留意されたい。しかしながら、実際には、個々のデベロッパシステム62は、図2のデベロッパエンティティシステム22のデベロッパエンティティによって所有され得るかまたは当該デベロッパエンティティに関連付けられ得る。したがって、個々のデベロッパシステム62は、図2のデベロッパエンティティシステム22のうちの1つを表わしている可能性もある。

#### 【0073】

この例示的な実施形態においては、個々のデベロッパシステム62は、付随するピア・ツー・ピア・エコシステムのノードと、個々のソフトウェアコンポーネントコントリビュータおよび/またはコンシューマのシステムとの両方として機能する。なお、本実施形態においては、デベロッパシステム62の個々のデベロッパはまた、単に、ローカルコンポーネントカタログ36への/からのソフトウェアコンポーネントのダウンロード動作および/またはアップロード動作だけではなく、トランザクション登録を実現する目的で、カタログインスタンス32および関連する機能にアクセスできるレジストラとしても機能することに留意されたい。

#### 【0074】

例示的なデベロッパシステム62は、UIディスプレイスクリーン70と通信するカタログインスタンス(この明細書中においては第1のカタログインスタンスまたはカタログインスタンス1と称される)32を含む。カタログインスタンス32はさらに、付随するピア・ツー・ピア・エコシステムをホストするネットワークまたはクラウド14上で実行される分散型ブロックチェーン台帳46、20と通信する。

#### 【0075】

説明の便宜上、UIディスプレイスクリーンは、ディスプレイ上に提供される如何なるソフトウェア生成描写であってもよい。描写の例は、ウィンドウ、ダイアログボックス、表示されたテーブル、および他のグラフィカルユーザインターフェイス特徴、たとえば、ブラウザなどのソフトウェアを介してユーザに提示されるユーザインターフェイス制御などを含む。単一の境界内に含まれているユーザインターフェイスディスプレイスクリーンはビューまたはウィンドウと称される。ビューまたはウィンドウは、サブビューまたはサブウィンドウ、ダイアログボックス、グラフ、テーブルなどのセクションを含み得る。いくつかの場合には、ユーザインターフェイスディスプレイスクリーンは、ディスプレイ上

10

20

30

40

50

にその時点で表示されているすべてのアプリケーションウィンドウを指していることもある。

【0076】

UI制御は、ユーザがデータを入力および閲覧すること、および/またはユーザインターフェイスと対話すること、を可能にするように適合されたユーザインターフェイス・ディスプレイスクリーンのうち表示された任意の要素またはコンポーネントであってもよい。ユーザインターフェイス制御の追加の例は、ボタン、ドロップダウンメニュー、メニューアイテム、タップアンドホールド機能などを含む。同様に、ユーザインターフェイス制御信号は、ソフトウェアのための入力として提供される如何なる信号であってもよい。この場合、入力は、ソフトウェアに関連付けられたユーザインターフェイスディスプレイスクリーンおよび/またはソフトウェアアプリケーションに影響を及ぼす。

10

【0077】

例示的なカタログインスタンス32はローカルの台帳レプリカ34を含む。ローカルの台帳レプリカ34は、分散型ブロックチェーン20のうち、複製された(他のカタログインスタンスまたはノード間で複製された)未検証の待ち行列38およびブロックチェーンレプリカ40を含む。カタログインスタンス32はさらに、(たとえば、リレーショナルデータベースまたは他のタイプのデータベースによって実現され得る)ローカルコンポーネントカタログ36、UIコード生成モジュール60、台帳インターフェイス型モジュール64、セキュリティおよび許可モジュール62、ならびにブロック演算モジュール66を含む。カタログインスタンスコントローラ68はカタログインスタンス32のさまざまな他のモジュール34、36、60~66の間の相互通信を接続し、制御し、ルーティングすることを容易にする。

20

【0078】

例示的なUIディスプレイスクリーン70ならびに付随するUI制御およびオプションは、コントリビュータおよびコンシューマの登録を容易にするためのUI制御;コンポーネント登録;ローカルコンポーネントカタログ36において利用可能なソフトウェアコンポーネントについてのコンポーネントサーチおよび/またはビュー;コンポーネント保水性、コンポーネント識別などの検証;ソフトウェアコンポーネントのソース、由来、ライセンス条件、サポート状態などの確認;デベロッパエンティティネームスペースの登録;コントリビュータ登録;コンポーネントのアップロードおよび/またはダウンロードの登録;ソフトウェアコンポーネントを登録解除するためのUI制御またはオプション;などを含む。

30

【0079】

なお、例示的なUIディスプレイスクリーン70を介して利用可能なUI制御の正確な数およびタイプは実現例特有のものであって、本教示の範囲から逸脱することなく所与の実現例のニーズに応じて異なり得ることに留意されたい。さらに、UI制御およびオプション70のうち1つ以上は、(たとえば、ウィザード形式で)複数のUIディスプレイスクリーンによって実現されてもよい。

【0080】

この例示的な実施形態においては、UIディスプレイスクリーン70のさまざまなUI制御およびオプションは、カタログインスタンス32のUIコード生成モジュール60からのレンダリング命令に基づいて生成される。

40

【0081】

セキュリティおよび許可モジュール62は、個々のデベロッパシステム62のデベロッパエンティティによって確立されたルールを考慮して、個々のユーザおよびデベロッパエンティティの認証および許可を容易にし得る。セキュリティおよび許可モジュール62はさらに、さまざまなセキュリティおよび許可機能の実現を容易にするために、図2の1つ以上のアイデンティティドメイン44と通信してもよい。なお、セキュリティおよび許可モジュール62によって実行される正確な機能が実現例特有のものであって、所与の実現例のニーズに応じて異なり得ることに留意されたい。

50

## 【 0 0 8 2 】

カタログインスタンス 3 2 のブロック演算モジュール 6 6 は、分散型台帳および付随するコンピューティングクラウドおよびエコシステム 1 4 の分散型ブロックチェーン 2 0 に含めるようにトランザクションの次のブロックを計算する（すなわち、検証する）ために、カタログインスタンス 3 2 が（たとえば、合意アルゴリズムに従って、またはラウンドロビン選択によって）選択されている場合に、ローカルの未検証の待ち行列 3 8 におけるペンディングトランザクションについてのハッシュの演算を容易にするための、かつ、ブロックチェーン 4 0、2 0 のブロックについてのハッシュを演算するためのコードを含む。

## 【 0 0 8 3 】

台帳インターフェイス型モジュール 6 4 は、未検証の待ち行列 3 8 を、他のカタログインスタンス（すなわち、エコシステムのノード）の未検証の待ち行列 4 6 に入力する新しいトランザクションを伝搬することを容易にするためのコードを含む。台帳インターフェイス型モジュール 6 4 はさらに、分散型ブロックチェーン 2 0 を複製するとともにブロックチェーン 2 0 の最新レプリカ 4 0 を維持するためのコードを含む。

## 【 0 0 8 4 】

ローカルコンポーネントカタログ 3 6 は、認可されたコンシューマによって選択的にダウンロードされ得るソフトウェアコンポーネントを維持する。ソフトウェアコンポーネントは、登録済みのデベロッパエンティティネームスペースおよび対応する登録済みデベロッパエンティティに関連付けて、認可済みおよび登録済みのコンピュータによってローカルコンポーネントカタログ 3 6 にアップロードされてもよい。

## 【 0 0 8 5 】

図 4 は、第 2 の例示的システム 8 0、デベロッパエンティティを登録するための付随するカタログインスタンスプロセス 9 4、および付随するデベロッパエンティティ登録トランザクション 8 4 を、図 1 ~ 図 3 の分散型ブロックチェーン台帳 2 0 と共に示す。この場合、第 2 の例示的システム 8 0 は図 1 ~ 図 3 の実施形態に従って実現されてもよい。

## 【 0 0 8 6 】

例示的なシステム 8 0 においては、たとえば、図 2 のレジストラシステム 4 2 のユーザーに対応するレジストラ 8 2 は、レジストラシステム 4 2 と、図 3 の付随する UI ディスプレイスクリーンおよびオプション 7 0 とを用いて、デベロッパエンティティ登録トランザクション 8 4 をサブミットする。デベロッパエンティティ登録トランザクション 8 4 は、（たとえば、図 2 のデベロッパエンティティシステム 2 2 のデベロッパエンティティに対応する）デベロッパエンティティ名 8 6、デベロッパエンティティの公開鍵 8 8、およびデベロッパエンティティのデジタル署名 9 0 を含む。

## 【 0 0 8 7 】

なお、デジタル署名 9 0 が公開鍵インフラストラクチャ（PKI）に従って実現され得るとともに、登録されているデベロッパエンティティの秘密鍵を含み得ることに留意されたい。デジタル署名は、（たとえば、アイデンティティドメインおよび付随する認証機関、たとえば図 2 のアイデンティティドメイン 4 4 によって）認可された（すなわち、分散型ブロックチェーン台帳 2 0 および付随の分散型エコシステムに登録することが認可された）デベロッパエンティティの代わりに、レジストラ 8 2 が登録をサブミットしていることの確認を容易にする。

## 【 0 0 8 8 】

この例示的な実施形態においては、レジストラ 8 2 は、第 1 のカタログインスタンス（たとえば、図 3 のカタログインスタンス 3 2）に対してデベロッパエンティティトランザクション 8 2 をサブミットする。第 1 のカタログインスタンス 3 2 は、次いで、最初のデベロッパエンティティ登録プロセス 9 4 を実行してから、選択されたカタログインスタンス（たとえば、第 1 のカタログインスタンス 3 2、第 2 のカタログインスタンス 1 1 0 または第 3 のカタログインスタンス 1 1 2）による分散型ブロックチェーン 2 0 に対する最終的なコミット 1 1 4 のために他のカタログインスタンス 1 1 0、1 1 2 の未検証の待ち

10

20

30

40

50

行列に対してトランザクション 84 を伝搬する。選択されたカタログインスタンス（たとえば、第 3 のカタログインスタンス 112）は、そのブロックチェーン演算モジュール（たとえば、図 3 のモジュール 66）を用いて、トランザクションハッシュおよびブロックハッシュを演算することにより、分散型ブロックチェーン台帳 20 に対してトランザクション 84 をコミットしてもよい。

#### 【0089】

最初のデベロッパエンティティ登録プロセス 94 は、分散型ブロックチェーン台帳 20 のトランザクションを後でコミットするためにカタログインスタンスに対してトランザクション 84 をサブミットする前に、デベロッパエンティティ登録トランザクション 84 の前処理の形式として、さまざまなステップ 98 ~ 108 を含む。最初の署名検証ステップ 98 は、たとえば、アイデンティティドメイン（たとえば、図 2 のアイデンティティドメイン 44）の認証機関（CA）を参照するデジタル署名 90 の検証を含む。

10

#### 【0090】

デジタル署名 90 が検証された後、プロセス 94 は、同じデベロッパエンティティ名 86 の既存のデベロッパエンティティ登録のいずれについても、分散型ブロックチェーン 20 および / または任意の付随する未検証の待ち行列 106 をチェックする。他の同様のデベロッパトランザクションが予め登録されている（かまたは、他の場合には、その時点で未検証である待ち行列に存在している）場合、現在のデベロッパエンティティ登録プロセス 94 は、不一致の検出により中断される。

#### 【0091】

次に、オプションのコミュニティリード検証ステップ 102 は、（エコシステムの）オプションのコミュニティリードで、コミュニティリードが分散型ブロックチェーン 20 へのデベロッパエンティティトランザクション 84 の登録に異議を有しているかどうかをチェックするステップを含む。コミュニティリードは、任意には、コミュニティリードがトランザクションをブロックすることが認可（または許可）されなければならないという何らかの理由で、デベロッパエンティティ登録トランザクション 84 をブロックしてもよい。

20

#### 【0092】

オプションのコミュニティリード検証ステップ 102 の後、登録トランザクション 84 は、たとえばトランザクション生成ステップ 104 によって、未検証の待ち行列に含めるためにパッケージされてハッシュされる。なお、ステップ 98 ~ 102 のうちいずれかが終了しない場合にはデベロッパエンティティトランザクション登録プロセス 94 が中断されることに留意されたい。

30

#### 【0093】

デベロッパエンティティ登録トランザクション 84 は、ローカルの未検証の待ち行列 38 にサブミットされた後、伝搬ステップ 108 において、コミット処理 114 の準備のために、他のカタログインスタンス 110、112 および付随する未検証の待ち行列（たとえば、図 3 の待ち行列 46）に伝搬される。これにより、未検証の待ち行列 38 および対応するレプリカ（たとえば、図 2 のレプリカ 46）における 1 つ以上のペンディングトランザクションの集合が、分散型ブロックチェーン 20 のうち 1 つ以上のブロックとしてコミットされるために処理される。

40

#### 【0094】

なお、この例示的な実施形態においては、複数のエンティティが 1 つ以上のクラウドアイデンティティドメイン（たとえば、図 2 のドメイン 44）によってプロビジョニングされているので、所与のカタログインスタンス 32、110、112 がシステム 80 のすべてのユーザおよびパーティシパントエンティティについての情報を識別するために既にアクセスしていたかもしれないことに留意されたい。

#### 【0095】

なお、図 2 のクラウドアイデンティティドメイン 44 の使用が、特定の実現例において省略され得る範囲において、任意であることに留意されたい。たとえば、代替的には、ユ

50

ーザおよびエンティティは、クラウドアイデンティティドメイン以外の他の手段によって識別され得る。さらに、他の実施形態においては、特定のユーザが識別されないままであるかまたは匿名のままであってもよい。

【0096】

しかしながら、この例においては、各々のユーザおよびエンティティは、図2のアイデンティティドメイン44内における予め定められたアイデンティティに関連付けられている。

【0097】

この例示的な実施形態は、個々のデベロッパ役割に関連付けられた許可と、個々の組織を表しているデベロッパエンティティに関連付けられた許可とを識別して関連付ける。

【0098】

複数の個体が単一のエンティティ（たとえば、企業）のためのコンポーネントを開発し得る可能性があり、このため、複数の個体のアイデンティティは、ソフトウェアコンポーネントの所有権および有効性の検証の目的で用いられ得る特定のデベロッパエンティティアイデンティティに（1つ以上のアイデンティティドメインにわたって）マッピングされるかまたは関連付けられることとなる。

【0099】

しかしながら、分散型ブロックチェーン20は、デベロッパエンティティについての識別子と公開鍵との間のマッピングまたは関連付けを記録する。この例示的な実施形態においては、格納されたアイデンティティは、デベロッパエンティティのアイデンティティを表わしている。ソフトウェアコンポーネントをカタログインスタンスに書込むおよび/またはアップロードする（とともに、コンポーネントおよび当該アップロードをブロックチェーン20に登録する）特定の個体はコントリビュータと称される。

【0100】

したがって、デベロッパエンティティ登録プロセス94は、デベロッパエンティティの自己登録を表わしている。レジストラ82は、システム80にアクセスできるとともにデベロッパエンティティの登録を希望している如何なるユーザであってもよい。

【0101】

したがって、（特定の實現例の需要に応じて異なり得る）例示的なデベロッパエンティティ登録プロセス94は、レジストラ82を任意に認証した後、（たとえば、図3のUI制御およびオプション70間に含まれ得るような）「デベロッパエンティティ」UIおよび付随のUI制御の「作成」を用いて、デベロッパエンティティ（たとえば、少なくとも識別名）についての情報84、ただし場合によってはコンタクト情報などを供給することを含む。付加的には、レジストラ82はまた、公開鍵88を供給し、トランザクションにデジタル署名することにより、たとえば、結果として、主張された鍵（たとえば、デベロッパエンティティ秘密鍵）を備えたデジタル署名90が得られることとなる。

【0102】

カタログインスタンスプロセス94は、次いで、（ステップ98において）供給された公開鍵を用いて署名90を検証し、これにより、レジストラ82が機密の片割れ半分を両方（たとえば、登録されているデベロッパエンティティに関連付けられた公共鍵および秘密鍵の両方）とも所有していることを証明する。次に、（ステップ100において）カタログインスタンス32は、関連するカタログデータストア（たとえば、図3のローカルコンポーネントカタログ36、ならびに/または、図3のローカルの未検証の待ち行列38および/もしくはローカルのブロックチェーンレプリカ40）をチェックして、デベロッパエンティティがまだ規定されていないかまたは規定されるプロセス中であることを確実にする。

【0103】

任意には、コミュニティリード検証ステップ102は、デベロッパエンティティトランザクション84に対応する要求を検証する。たとえば、特定の實現例においては、追加の手作業の工程（たとえば、署名すべき書類事務）は、分散型ブロックチェーン20に対す

10

20

30

40

50

るトランザクション 8 4 のコミットよりも前に、デベロッパエンティティ登録の前段階プロセスを完了する必要があるかもしれない。最後に、登録トランザクションは、(ステップ 1 0 4 において)作成および/またはフォーマットされて、(ステップ 1 0 8 において)他のカタログインスタンス 1 1 0 および 1 1 2 に伝搬される。

#### 【0104】

伝搬後、所与の登録トランザクションは、分散型ブロックチェーン 2 0 にコミットされるまで、未検証の待ち行列(たとえば、図 3 の未検証の待ち行列 4 6)のレプリカに位置することとなるだろう。作成されたトランザクションは、信頼されていない(すなわち、未検証の)トランザクションとしてエコシステムの他のすべてのフルノード(たとえば、カタログインスタンス 1 1 0 および 1 1 2)に伝搬される。未検証の待ち行列 3 8 は、未検証の待ち行列の未検証のトランザクションがブロックチェーン 2 0 にコミットされるまで、対象である未検証のトランザクションに配置することが可能であり得る。

10

#### 【0105】

分散型ブロックチェーン 2 0 に対するデベロッパエンティティトランザクション 8 4 のコミット後、関連するデベロッパエンティティは、分散型ブロックチェーン 2 0 にその時点でコミットされているデベロッパエンティティ登録トランザクション 8 4 において維持されているとおりに、デベロッパエンティティのアイデンティティに依拠する追加のトランザクションの開始を可能にするのに十分なほどに信頼されていると見なされる。

#### 【0106】

要約すると、図 4 は、デベロッパエンティティのアイデンティティ(すなわち、デベロッパアイデンティティ)のために台帳記入を作成するためのシステム 8 0 内の基本コンポーネントを例示している。

20

#### 【0107】

システム 8 0 が示しているレジストラ 8 2 は、コンピュータシステム 4 2 および付随するカタログインスタンスユーザインターフェイスを用いて、デベロッパエンティティトランザクション情報 8 4 によって示されているようにデベロッパエンティティを規定する。レジストラ 8 2 は、このプロセスの前に任意に認証することができる。

#### 【0108】

レジストラシステム 4 2 のユーザインターフェイスは、レジストラ 8 2 が、識別名 8 6 およびセキュリティメカニズム(たとえば、公開鍵 8 8 およびデジタル署名 9 0)などのデベロッパアイデンティティについての情報 8 6 ~ 9 0 を供給することを可能にする UI 制御およびオプションを含む。

30

#### 【0109】

なお、レジストラ 8 2 が人、ビジネスエンティティ、自動化プロセスなどであり得ることに留意されたい。人がレジストラ 8 2 である場合、デベロッパエンティティ登録を実現するためのユーザインターフェイスは、(デスクトップパソコン、ラップトップ、タブレット、スマートフォンまたは他のデバイスであり得る)レジストラシステム 4 2 を介してアクセス可能である。

#### 【0110】

さまざまな実施形態においては、ユーザインターフェイスおよび付随のレジストラシステム 4 2 は、全体的または部分的には、ネットワークを介して連結されるサーバ、クラウドサービスなどの遠隔設置されたプロセッサによって、処理することができる。レジストラの機能が自動的に実行される場合、プロセスは、任意の好適な処理ハードウェアおよびソフトウェアを用いてそれら機能を実行することができる。一般には、特に規定のない限り、この明細書中に記載される機能は、さまざまな地理的領域において、さまざまな時点に位置するハードウェアとソフトウェアとの任意の組合せによって実行することができる。

40

#### 【0111】

デベロッパエンティティトランザクション 8 4 を含む情報 8 6 ~ 9 0 は、(この例示的な実施形態においてカタログプロセスとも称される)デベロッパエンティティ登録プロセ

50

ス 9 4 を実現するための、第 1 のカタログインスタンス 3 2 に対する要求によって転送される。

【 0 1 1 2 】

カタログプロセス 8 4 内に示されるように、ステップ 9 8 ~ 1 0 8 は、供給された公開鍵 8 8 を用いて署名 9 0 を検証するステップ、(同じエンティティが既に規定されている場合には)既存のデベロッパエンティティについてチェックするステップ、および、任意には、コミュニティリードに検証(たとえば、手作業で署名する書類事務)を実行させるステップを含む。これらのステップのうちのいずれかに失敗すると、プロセスが中断される。それ以外の場合、トランザクション 8 4 はフォーマットされて、未検証の待ち行列 3 8 に組込まれるようにサブミットされる。

10

【 0 1 1 3 】

カタログプロセス 9 4 は、対象のレジストラ 8 2、デベロッパエンティティまたは他の実行者から遠隔で実行することができる。たとえば、カタログプロセス 9 4 および他のカタログプロセスの機能は、ピア位置などにあるサーバコンピュータ、クラウドサービス、パーソナルコンピュータシステム上で実行することができる。

【 0 1 1 4 】

追加のカタログインスタンス 1 1 0 および 1 1 2 は、第 1 のカタログインスタンス 3 2 のピアの存在を示している。ステップ 1 0 2 でフォーマットされたトランザクション 8 4 は、未検証の状態ですれら 1 1 0、1 1 2 に伝搬される。最終的に、十分な未検証のトランザクションが待ち行列 3 8 内に存在している場合、これらトランザクションが検証されて、これらのカタログインスタンス 3 2、1 1 0、1 1 2 のうちの 1 つによって(ブロックチェーン 2 0 のブロックとして)台帳にコミットされることとなる。検証されたトランザクション、すなわちデベロッパエンティティの定義は、この場合、カタログインスタンス 3 2、1 1 0、1 1 2 すべてがアクセス許可を読取った分散型ブロックチェーン 2 0 を介して、これらすべてのカタログインスタンス 3 2、1 1 0、1 1 2 にとって利用可能になるだろう。トランザクションがコミットされると、カタログインスタンス 3 2、1 1 0、1 1 2 はすべて、複製のデベロッパエンティティ登録(すなわち、複数のトランザクション)の生成を防止することとなる。いくつかの実施形態においては、カタログインスタンス 3 2、1 1 0、1 1 2 は、最初のトランザクションが未検証の状態であっても、複製を防止し得る。

20

30

【 0 1 1 5 】

デベロッパエンティティがエコシステム 8 0 上で登録されて検証されたときには、デベロッパエンティティはフルノード 3 2、1 1 0、1 1 2 すべてに伝搬されてしまっていることになるだろう。その後、1 つ以上のネームスペースがデベロッパエンティティに割付けられてもよい。ネームスペース登録プロセスは、デベロッパエンティティ機密を所有しているいずれの個体によっても開始することができるという点で、最初のデベロッパエンティティ登録プロセスと同様であり得る。

【 0 1 1 6 】

カタログエントリおよびインスタンスのピア・ツー・ピア伝搬および検証は、仮想通貨実現例(たとえばBitcoin、Dogecoinなど)において用いられる共有台帳のアプローチと同様であり得る。しかしながら、仮想通貨のシステムとは対照的に、ブロックの生成は演算集約的にされる必要はない。なぜなら、企図されるソフトウェアコンポーネント・エコシステム 8 0 が、要望に応じて(たとえば、図 2 の例示的なアイデンティティドメイン 4 4 によって)認証することができる既知のエンティティを含んでいるからである。したがって、(仮想通貨アプリケーションにおいて「マイニング(mining)」と称されることもある)時間のかかる演算要件は不要となる。

40

【 0 1 1 7 】

図 5 は、図 4 の第 2 の例示的システムおよび付随するエコシステム 8 0 によって実現可能であるとともに図 1 ~ 図 3 の関連するエコシステムによっても実現可能である第 1 の例示的な方法 1 2 0 を示すフロー図である。

50

## 【0118】

第1の例示的な方法120は、たとえば、図4のレジストラ82によって、デベロッパエンティティ名、公開鍵およびデジタル署名をカタログインスタンス（たとえば図4のカタログインスタンス32）に対してサブミットするステップを含む、追加情報サブミットステップ122を含む。

## 【0119】

次に、（たとえば、図4のデベロッパエンティティ登録トランザクション情報84に対応する）サブミットされた情報は、検証ステップ124において検証され、次いで、情報がブロックチェーンおよび/または未検証の待ち行列において複製のエントリを表わしているかどうかチェックされる。これらのチェック124および126がともにパスした場合、関連するデベロッパエンティティ登録トランザクションがステップ128において未検証の待ち行列の上に配置されて、ステップ130において他のカタログインスタンスに伝搬される。それ以外の場合、トランザクション登録プロセスが処理は異常終了する。

10

## 【0120】

他のカタログインスタンス（すなわち、ノード）に対するトランザクションの伝搬の後、これらノードは、他のカタログインスタンスに対するトランザクションを伝搬しているカタログインスタンス（たとえば、図4のカタログインスタンス32）についてピア資格条件が満たされているかどうかを集約的に判断してもよい。登録を実行するカタログインスタンスが（所与の実現例のニーズに応じて異なり得る）予め定められたピア資格条件を満たさない場合、登録プロセスは失敗となる。それ以外の場合、利用可能なカタログインスタンスのうちカタログインスタンスは、分散型ブロックチェーン台帳（たとえば、図4のブロックチェーン20）に対してトランザクションをコミットするのに必要とされる演算（たとえば、ハッシュの演算）を実行するためにステップ134において選択される。

20

## 【0121】

1つ以上のトランザクションのグループを検証済み待ち行列から分散型ブロックチェーン（たとえば、図4のブロックチェーン20）に転送するために演算を実行するための所与のカタログインスタンスを選択するのに用いられるアルゴリズムまたは方法は、実現例特有のものであるとともに、所与の実現例のニーズに応じて異なり得る。たとえば、特定の実施形態においては、第1の利用可能なカタログインスタンス（または最も利用可能なコンピューティングリソースを用いたインスタンス）は、演算を実行するためにカタログインスタンスとして選択されており、他の実現例においては、カタログインスタンスは、どのカタログインスタンスが演算を実行すべきであるかを判断するためにカタログインスタンスの多数決を取ってもよく、さらに他の実現例においては、カタログインスタンスは、（ラウンドロビンアプローチによって）ブロックを順番にコミットしてもよい。

30

## 【0122】

要約すると、図5は、図4のシステム80によって実現され得る方法120のフローチャートを示す。ステップ122は、図4の分散型ブロックチェーン台帳20にコミットされるべき新しいトランザクションとして集合的に提案されているデベロッパエンティティ名、公開鍵およびデジタル署名を受取るために実行される。

40

## 【0123】

ステップ124においては、デジタル署名を検証するためにチェックがなされる。検証が失敗すると、プロセスが障害状態で終了し、トランザクションは台帳にコミットされない。それ以外の場合、署名が検証をパスする場合、ステップ126に進んで、複製のエントリ（たとえば、先に用いられていた名前）についてチェックがなされる。チェック126が失敗すると、上述のようにプロセス120は異常終了する。それ以外の場合、待ち行列にトランザクションを配置するためにステップ128が実行される。次に、ステップ130において、トランザクションが、ネットワーク上の他のピアプロセス（すなわち、ピア・ツー・ピア・コンポーネント・エコシステムの一部である）にとって利用可能にされる。

50

## 【0124】

ステップ132および134は、基礎をなす台帳のルールに従ってトランザクションを含むブロックを検証し、次いで、トランザクションがコミットされた分散型ブロックチェーン台帳（たとえば、図4のブロックチェーン20）によって、トランザクション情報をコミットしてすべてのカタログインスタンスにわたって共有する。

## 【0125】

図6は、UIディスプレイスクリーン70を介する図3のユーザデベロッパコンピューティングシステム62などのコンピューティングシステム上に表示され得る例示的なUIディスプレイスクリーン140を示す。例示的なUIディスプレイスクリーン140は、ソフトウェアコンポーネントのカタログをブラウズしたり、（たとえば、サーチボタン146によって）検索したり、（たとえば、ダウンロードボタン148によって）ダウンロードしたり、（たとえば、編集ボタン150によって）編集したり、付随するコンピューティングシステムがアクセスするカタログインスタンス（たとえば、図3のカタログインスタンス32）によって維持されるカタログエントリ/オプション144からソフトウェアコンポーネント156、158、186、188を（たとえば、削除ボタン152によって）削除したりするための、さまざまなUI制御およびオプション144~180を含む。

10

## 【0126】

この例示的な実施形態においては、カタログ144におけるソフトウェアコンポーネントは、特徴とされるコンポーネント160およびプライベートコンポーネント180に分割される1つ以上の対応するコンポーネントカード156、158、186および188によって示される。

20

## 【0127】

ソフトウェアコンポーネントの各々は、コンポーネント名162、172、192および202、対応するファイル名164、174、194および204、ならびに記述166、176、196、206を含む。所与のコンポーネントUIカード156、158、186、188がユーザによって選択されてもよい。カードを右クリックすると追加のユーザオプションが現われ得る。これら追加のユーザオプションは、たとえば、所与のコンポーネントについてより多くの情報のドリルダウンを行ったり、コンポーネントと対話する（たとえば、コンポーネントをダウンロードする）ために追加のユーザオプションおよび制御にアクセスしたりするためのものである。なお、カードの右クリックとは対照的に、「詳細ボタン」の選択などの他のUIアフォーダンスが、本教示の範囲から逸脱することなく用いられてもよい。

30

## 【0128】

要約すると、例示的なUIディスプレイスクリーン140はカタログUIのうちの例示的な一部分を表わしている。UIディスプレイスクリーン140は、ソフトウェアコンポーネントについての情報を含むさまざまな例示的なUIカードまたはサムネイル156、158、186、188を示している。

## 【0129】

なお、各々のコンポーネントカード156、158、186、188が追加情報を示し得ること、たとえば、グラフ図の情報または画像（たとえば、コンポーネントが生成するUIディスプレイスクリーンの画像）、著者情報、ネームスペース指定、コンポーネントについてのさらに詳細な説明、所与のコンポーネントの状態（たとえば、開発サイクルにおけるサポート状態または位置）、コンポーネントがダウンロードされた回数、コンポーネントがデモであるか否か、などを含み得ることに留意されたい。他の実施形態においては、他のタイプの情報を提供することに加えて、コンポーネントを表わすためのさまざまな方法を用いることができる。

40

## 【0130】

例示的なコンポーネントUIカード156の場合、名前は「ora-card-layout」と示される。この例示的な方式における名前の最初の部分「ora」は、基礎の

50

「ネームスペース」と称される。その後ろに、完全かつ固有のコンポーネント名を形成するための追加のテキストを追加することができる。

【0131】

なお、コンポーネントについてのネームスペースも、図6に示されるコンポーネント名接頭辞アプローチと同様に、またはこの代わりに、いくつかのシステムにおけるコンポーネントメタデータ中のどこかに記録され得ることに留意されたい。しかしながら、どちらのアプローチにおいても、関連するエコシステムにおける各コンポーネントごとに最終的に固有となるネームスペースおよびコンポーネント名が、ローカルのソフトウェアコンポーネントカタログの利用可能なソフトウェアコンポーネントを示す所与のUIディスプレイスクリーンに示される。エコシステムは、いくつかの実施形態においては、所与のいずれかのコンポーネントについての固有のアイデンティティを規定するために、ネームスペースおよびコンポーネント名に加えて、合成アイデンティティ、たとえば、128ビットのユニバーサルユニーク識別子(Universal Unique Identifier: UUIID)、を割付け

10

【0132】

各々のデベロッパエンティティには基礎のネームスペースが割当てられる。基礎のネームスペースは、1つ以上の用語であり得、その後ろに、デベロッパが選択する際の追加の記述子または名前が追加され得る。(たとえば、コンポーネントUIカード186のファイル名194に対応する)名前「ojx」は、図6に示されるコンポーネントにおける基礎のネームスペース指示子の別の例である。なお、図6の例においては、基礎のネームスペースが3文字であるが、他の実施形態においては、如何なる数またはタイプの指示を用いてもよいことに留意されたい。

20

【0133】

なお、さまざまなコンポーネントUIカード156、158、186、188が、各々の識別されたコンポーネントに関連付けられたブロックチェーン登録情報に関する追加情報を示し得ることに留意されたい。たとえば、検証されてブロックチェーン台帳にコミットされた情報(たとえば、コンポーネントを供給する際に用いられるコンポーネント名、ネームスペース、所有者、状態およびライセンス情報も閲覧可能である。

【0134】

この例示的な実施形態においては、カタログUIは、付随するブロックチェーントランザクションにおけるそのコンポーネントについての対応する登録のチェックサムと一致するチェックサムを有していることなどによって、検証されるダウンロードすべきソフトウェアコンポーネントについてのUIカードを表示するだけであるだろう。

30

【0135】

なお、ユーザがたとえばコンポーネントをダウンロードなどの動作を実行する場合、ダウンロードプロセスは、そのコンポーネントのためのダウンロード登録プロセスの完了を含み得ることに留意されたい。さらに、例示的なUIディスプレイスクリーン140に示されるもの以外の追加のUI制御およびオプションが実現可能であることに留意されたい。たとえば、さまざまなタイプのブロックチェーン登録トランザクションなどを開始する、コンポーネントをアップロードするための追加のオプションが提供されてもよい。

40

【0136】

図7は、第3の例示的なシステム210によって表わされるピア・ツー・ピア・ソフトウェアコンポーネント・エコシステムの(カタログインスタンスと称される)相互に通信するノード214~218によって維持されるブロックチェーン212および230の初期状態を例示する(ソフトウェアコンポーネント・エコシステムを表わす)第3の例示的なシステム210を説明する第1のブロック図である。図7は、空のルートブロック212(ブロック0)を備えた(ブロックチェーンのブロック212の分散型レプリカ230~236として明示する)ブロックチェーンを含むシステム210の例示的な初期状態を示す。

【0137】

50

この例示的な実施形態においては、3つのカタログインスタンス214～218は、さまざまなコンピュータシステムおよび/または仮想マシン上で実行されるソフトウェアを表わす。この場合、コンピュータシステムおよび/または仮想マシンはネットワークを介して相互に通信する。カタログインスタンス214～218は、1つ以上のクラウド（たとえば、1つ以上の相互通信サーバの集合）によって実現されてもよい。

#### 【0138】

カタログインスタンス214～218の各々は未検証の待ち行列224～226と共に初期状態で示されている。説明の便宜上、未検証の待ち行列は、コンピューティングオブジェクトが処理されてブロックチェーン230～236に含まれる前に、コンピューティングオブジェクト（たとえば、1つ以上のトランザクションに対応するコンピューティングオブジェクト）を格納するための如何なるデータストレージメカニズムであってもよい。なお、ブロックチェーン230～236が、カタログインスタンス214～218間で共有されるかまたは複製されるブロックチェーンの個々のレプリカを含むことに留意されたい。

10

#### 【0139】

カタログインスタンス230～226の内容212の共有は、各々のカタログインスタンス214～218によって実現されるソフトウェアによって実現されてもよい。この場合、ソフトウェアは、それぞれのブロックチェーン230～236間でのブロックチェーン内容212の複製を含む、カタログインスタンス214～218間の通信を可能にするためのコンピュータコードを含む。なお、ブロックチェーン230～236がその時点で最初のルートブロック（最初の端末ブロック、ブロック0またはゼロ番目のブロックとも称される）212だけを含んでいることに留意されたい。最初は、最初の端末ブロック212には、他の場合にブロック212のペイロードセクション220に格納される可能性もあるトランザクションが無い可能性がある。

20

#### 【0140】

なお、この明細書中に記載されるとともに以下においてより詳細に説明されるトランザクションペイロード、トランザクションハッシュ、トランザクション公開鍵、ブロックハッシュ、ブロックペイロードなどは単なる例示に過ぎず、この教示の範囲から逸脱しない程度に異なってもよい。さらに、実現可能なトランザクションのほんの小規模な例示的サブセットだけがこの明細書中においてより十分に例示および記載されている。この教示を利用できる当業者であれば、所与の実現例のニーズを満たすために、過度の実験を行うことなく、さまざまなエコシステムノードによって選択的にアクセスできるようにブロックチェーンに収集、検証、格納するために、この明細書中に記載される実施形態を容易に実現および使用し得る。

30

#### 【0141】

最初の端末ブロック212はこの使用事例においてはゼロである（前のハッシュと称される）前のブロックのハッシュのためのフィールドを備えたハッシュセクション222を含む。ハッシュセクション222はさらに、例示の目的で、#2ab4と示される現在のハッシュ（ブロックチェーンのための開始ハッシュとも称される）を含む。なお、ハッシュ記号「#2ab4」の「#」は、SHA-256（Secure Hash Algorithm（セキュアハッシュアルゴリズム）-256）または他のタイプのハッシュであり得る、より長いハッシュ値についてのワイルドカード文字またはプレースホルダを表わし得ることに留意されたい。使用される正確なタイプのハッシュは実現例特有のものであってもよく、本教示の範囲から逸脱しない程度に異なってもよい。

40

#### 【0142】

図8は、第1の例示的トランザクション（すなわち、デベロッパ登録トランザクション）240が、本使用事例において第1のカタログインスタンス214である発生元カタログインスタンス（たとえば、第1のカタログインスタンス）によって、カタログインスタンス214～218の未検証の待ち行列224～226に伝搬された後における、図7の第3の例示的システム210を示す第2のブロック図である。

50

## 【 0 1 4 3 】

なお、所与のカタログインスタンス 2 1 4 ~ 2 1 8 が、所与のカタログインスタンス 2 1 4 ~ 2 1 8 に対するアクセスのために（たとえば、公開鍵インフラストラクチャ（PKI）によって）認証されたデベロッパエンティティ（または他のエンティティ）のデベロッパコンピュータ（デベロッパシステムと称される）と通信し得ることに留意されたい。カタログインスタンス 2 1 4 ~ 2 1 8 は、ソフトウェアコンポーネントのデベロッパ、コントリビュータおよび/またはコンシューマが、カタログインスタンス 2 1 4 ~ 2 1 8 での初期認証および登録中に供給されるクレデンシャルに関連付けられた特権に従って、認証済みの特定のカタログインスタンス 2 1 4 ~ 2 1 8 へのおよび/またはからのソフトウェアコンポーネントのアップロードおよび/またはダウンロードを実行することを可能にするためのUIディスプレイスクリーンを生成することを容易にするために、ソフトウェアを実行してもよい。

10

## 【 0 1 4 4 】

なお、所与のデベロッパ（および/またはコントリビュータ、コンシューマなどの）エンティティ（たとえば、企業）は、特定のデベロッパユーザおよび関連するコンピュータシステムをカタログインスタンス 2 1 4 ~ 2 1 8 の1つ以上にインターフェイス接続するために、レジストラエンティティを用いてもよい。たとえば、トランザクションのダウンロードおよび/またはアップロード、さまざまなタイプの登録の開始、などを実行するためにカタログインスタンス 2 1 4 ~ 2 1 8 と通信するデベロッパおよび関連するコンピュータシステムは、デベロッパが属するデベロッパエンティティのためのプロキシとして機能し得る。

20

## 【 0 1 4 5 】

図 8 に示されるように、この使用事例においては、第 1 のカタログインスタンス 2 1 4 は、ブロックチェーン 2 3 0 ~ 2 3 4 に登録されるべき第 1 のトランザクション 2 4 0 を受取っていた。第 1 のカタログインスタンス 2 1 4 は、第 1 のトランザクション 2 4 0 を作成して第 1 の未検証の待ち行列 2 2 4 に格納するとともに第 1 のトランザクション 2 4 0 のコピーまたはレプリカをエコシステムノード 2 1 4 ~ 2 1 8 の他の未検証の待ち行列 2 2 6 ~ 2 2 6 に伝搬することを容易にするためのコンピュータコードを含む。

## 【 0 1 4 6 】

第 1 の例示的なトランザクション 2 4 0 は、（たとえば、「レジスタデベロッパ」トランザクションタイプを指定する）タイプ情報と、デベロッパエンティティ情報（たとえば、「example.com」）とを含む第 1 のトランザクションペイロードセクション 2 5 0 を含む。なお、第 1 のトランザクション 2 4 0 のトランザクションペイロードセクション 2 5 0 に格納されたデータが単に例示的なものに過ぎず、本教示の範囲から逸脱しない程度に異なり得ることに留意されたい。

30

## 【 0 1 4 7 】

さらに、この教示の範囲から逸脱することなく、トランザクション 2 4 0 の他のセクションからの1つ以上のデータアイテム（たとえば、ハッシュセクション 2 5 2）が、ペイロードセクション 2 5 0 に含まれていてもよく、および/または、この逆も同様に可能であってもよい。第 1 の例示的なトランザクション 2 4 0 のハッシュセクション 2 5 2 は、第 1 のトランザクション 2 4 0 のデベロッパエンティティ登録者に関連付けられた第 1 のトランザクションハッシュ（たとえば、# 1 7 7 e）および公開鍵（たとえば、# e 5 6 c）を含む。

40

## 【 0 1 4 8 】

第 1 のデベロッパ登録トランザクション 2 4 0 の生成後、第 1 のカタログインスタンス 2 1 4 は、他のカタログインスタンス 2 1 6 および 2 1 8 の未検証の待ち行列 2 2 6 および 2 2 6 に最初に格納するために第 1 のトランザクション 2 4 0 を伝搬するためのピア・ツー・ピア・エコシステムソフトウェアおよびソフトウェアインターフェイス型ブロックチェーンを含む。なお、他のノード 2 1 6 および 2 1 8 が、ネットワークレイテンシなどに応じて、必ずしも同時に第 1 のトランザクション 2 4 0 を受取らない可能性もあること

50

に留意されたい。

【0149】

図9は、デベロッパ登録トランザクション240を処理（たとえば、検証または確認）するために（選択ルールまたはアルゴリズムに従って）選択されたカタログインスタンス（たとえば、第2のカタログインスタンス216）による、最初のデベロッパ登録トランザクション240の検証およびブロックチェーン230～236に対するコミットの処理中における、図8の第3の例示的なシステムを示す第3のブロック図である。

【0150】

なお、ブロックチェーン230～236のブロックに組込むべきトランザクション240の処理および検証をカタログインスタンス214～218のいずれが担うこととなるかを判断するのに用いられる選択ルールまたはアルゴリズムは、実現例特有のものであって、所与の実現例のニーズに応じて異なってもよいことに留意されたい。たとえば、特定の実施形態においては、第1の利用可能なカタログインスタンス214～218が処理のために選択される。他の実施形態においては、単一のカタログインスタンス214～218が、ブロックチェーンブロックへの組込みのためにすべてのトランザクションの処理専用に使われている。他の実施形態においては、すべてのカタログインスタンス214～218がトランザクションを同時に処理するとともに、最初に処理を完了すべきカタログインスタンス214～218が、結果として生じる検証済みブロックをブロックチェーン230～236に対してコミットする。他の実施形態においては、ブロックの処理を完了するのに必要な処理と労力が、カタログインスタンス214～218の各々に関連付けられた信頼性スコアまたはメトリックに従って割付けられてもよい。他の実施形態においては、どのカタログインスタンス214～218が1つ以上のトランザクションの所与のセットを処理すべきかをこれらカタログインスタンス214～218が判断することを可能にするためのコンセンサスまたは投票メカニズムが用いられる。

【0151】

第1のデベロッパ登録トランザクション240の（この明細書中においてトランザクション検証または確認と称される）処理中に、選択されたカタログインスタンス216は、最初に、第1のトランザクション240が組込まれるべきブロックについてのツリーハッシュを演算する。ツリーハッシュ（たとえばマークルツリー）は、（この場合、第1のデベロッパ登録トランザクション240だけを含む）すべてのペンディングトランザクションのハッシュを含んでいる。結果として生じるツリーハッシュ（たとえば、#DD34）は、以下においてより十分に記載されるように、第1のデベロッパ登録トランザクション240のために作成されたブロックに組込まれている。

【0152】

要約すると、ブロックチェーン230～236における新しいブロックに対するトランザクションのコミットに備えて、任意の顕著なトランザクションをロールアップするように、カタログインスタンスによって実現されるプラグ接続可能な検証機能に基づいて、1つ以上のノード（カタログインスタンス214～218）が割当てられる。

【0153】

ノードがいくつ含まれているかについての判断は実現例ごとに異なり得る。ノードがすべて信頼されている単純な実現例においては、単一のノードがトランザクション確認を担っている可能性がある。他の実現例においては、複数のノードが同時に検証を行なって、結果を比較してコンセンサスを得る可能性もある。トランザクション確認プロセスおよびブロック生成は、基礎をなすブロックチェーン実現例の詳細に基づいて異なってもよい。この例示的な実施形態においては、ハッシュ値（たとえば、T1ハッシュ#DD34）を計算するための難易度レベルを選択的に変える必要はない。しかしながら、特定のエコシステム内の全体的な信頼レベルに応じて、および/または、共有されたブロックチェーン230～236に対するトランザクション240のコミットを含むトランザクション確認を実行するために割当てられた個々のカタログインスタンス214～218に関連付けられた信頼レベルに従って、ブロックハッシュ演算難易度レベル（および/または他の

10

20

30

40

50

ブルーフオワーク (proof-of-work) の難易度レベル) がステーリングされ得ることに留意されたい。

【0154】

図10は、選択されたカタログインスタンス216によるブロックチェーン230~236のためのブロック260の(検証または確認とも称される)演算中における、図9の第3の例示的なシステム210を示す第4のブロック図であって、ブロック260は、図9のデベロッパ登録トランザクション240を含む未検証の待ち行列224~226にペンディングトランザクションを含んでいる(この場合、T1とも称されるトランザクション240を1つだけ含んでいる)。

【0155】

第2のカタログインスタンス216は、(T1ツリーハッシュ#DD34の演算後に)ブロック260の確認を完了する。ここでは、第1のブロック260についての第1のブロックハッシュ(#6ccf)を作成することを含む。ブロック1(B1)ハッシュとも称される第1のブロックハッシュ(#6ccf)は、前のブロックハッシュ(B0ハッシュ#2ab4)を選択的にハッシュすることによって、および/または、(所与の実現例のニーズを満たすために必要であり得るかまたは必要であり得ない)T1ツリーハッシュ#DD34および他の情報(たとえば、第1のデベロッパ登録トランザクション(T1)240に格納された情報)と組み合わせるかもしくは組込むことによって、演算される。結果として得られるB1ハッシュは、直前のブロック212のハッシュ(B0ハッシュ)と共に第1のブロック(B1)260に含まれている。

【0156】

なお、第1のブロック(B1)260が、第1のトランザクション240またはそのハッシュされたバージョンもしくは符号化されたバージョンを含むトランザクションセクション262を含むことに留意されたい。例示を目的として、B1ブロック(B1ハッシュ)#6ccfのためのハッシュと、前のブロック(B0ハッシュ)#2ab4のためのハッシュとを含むハッシュセクション266が示される。

【0157】

トランザクションセクション262はT1ツリーハッシュ#DD34を用いて符号化されてもよい。この例示的な実施形態においては、B1ブロックは、T1ツリーハッシュ#DD34、B1ハッシュ#6ccff、およびB0ハッシュ#2ab4を含む少なくとも3つのハッシュと、第1のトランザクション(T1)240のための公開鍵#e56cとを含むかまたは組込んでいる。B1ブロック260におけるT1トランザクション240はまた、図9に示されるように(T1ツリーハッシュ#DD34を演算するのに用いられる)T1トランザクションハッシュ#177eを組込んでいてもよく、および/または格納していてもよい。

【0158】

公開鍵#e56cはまたハッシュを表わしていてもよい。さらに、特定の実現例において、B1ハッシュ#6ccfが、B1ブロック260を符号化するかもしくはハッシュするために用いられてもよく、および/または、B1ブロック260を表すコンピューティングオブジェクトもしくは記録に単に含まれていてもよいことに留意されたい。

【0159】

(T1トランザクション240を符号化するのに用いられ得る)T1ツリーハッシュ#DD34を含むさまざまなハッシュ266の演算後、B1ブロックが検証されており、第2のカタログインスタンス(C2)216によるエコシステム210の他のカタログインスタンス214および218への伝搬の準備ができています。

【0160】

要約すると、第2のカタログインスタンス216は、T1ツリーハッシュ#DD34および関連するトランザクションを記録する新しいブロック260を作成する。新しく確認されたブロック260は、図11に示されるように、ブロックチェーン230~236の端末ブロックになる。

10

20

30

40

50

## 【 0 1 6 1 】

図 1 1 は、第 2 のカタログインスタンス 2 1 6 によって処理されたブロック 2 6 0 がブロックチェーン 2 3 0 ~ 2 3 6 にコミットされて、エコシステム 2 1 0 のカタログインスタンス 2 1 4 ~ 2 1 8 の各々によって維持されるブロックチェーンレプリカに伝搬された後における、図 1 0 の例示的なシステムを示す第 5 のブロック図である。なお、T 1 トランザクション 2 4 0 がこのとき検証されて、他のカタログインスタンス 2 1 4 ~ 2 1 8 に伝搬されたブロック 2 6 0 にコミットされているので、T 1 トランザクション 2 4 0 が未検証の待ち行列（すなわち、そのレプリカ）2 2 4 ~ 2 2 6 から除外されることに留意されたい。

## 【 0 1 6 2 】

図 1 2 は、第 2 のカタログインスタンスが第 2 のトランザクション（すなわち、第 1 のネームスペース登録トランザクション T 2 ）2 7 0 を開始して、ソフトウェアコンポーネント・エコシステム 2 1 0 における他のカタログインスタンス 2 1 4 ~ 2 1 8 のうち未検証の待ち行列 2 2 4 ~ 2 2 6 に対して第 2 のトランザクション T 2 2 7 0 を伝搬した後における、図 1 1 の例示的なシステム 2 1 0 を示す第 6 のブロック図である。

## 【 0 1 6 3 】

第 2 の例示的なトランザクション（T 2 ）2 7 0 は例示的なトランザクションペイロードセクション 2 7 2 を含む。例示的なトランザクションペイロードセクション 2 7 2 は、（たとえば、このトランザクション 2 7 0 がネームスペーストランザクションであることを示している）トランザクションタイプ情報と、T 2 トランザクション 2 7 0 に関連付けられたデベロッパエンティティ名（たとえば、example.com）と、T 2 トランザクション 2 7 0 に関連付けられたネームスペース登録に結び付けられるデベロッパエンティティ（example.com）に関連付けられたネームスペース情報（たとえば、1 つ以上の付加的な名前および / または用語の集合のために用いられ得る名前）とを含む。例示を目的として、T 2 トランザクション 2 7 0 のためのトランザクションハッシュ（T 2 ハッシュ # 4 a f 5 e）を備えたハッシュセクション 2 7 4 を含む T 2 トランザクションが示されている。

## 【 0 1 6 4 】

図 1 2 のこの使用事例においては、第 2 のカタログインスタンス 2 1 6 は、T 2 トランザクション 2 7 0 の生成と、他のカタログインスタンス 2 1 4 および 2 1 8 の未検証の待ち行列 2 2 4 ~ 2 2 6 へのその後の伝搬とを容易にする。

## 【 0 1 6 5 】

要約すると、ネームスペーストランザクション（T 2 ）2 7 0 が、デベロッパエンティティについての新しいネームスペース（たとえば、E C O M）を登録するために（たとえば、デベロッパエンティティ example.com のレジストラによって）サブミットされる。第 2 のカタログインスタンス 2 1 6 は、E C O M と称されるネームスペースについてのデベロッパエンティティによるネームスペース登録トランザクションを表わす T 2 トランザクション 2 7 0 を作成する。この T 2 トランザクション 2 7 0 は、エコシステム 2 1 0 にわたってブロードキャストされるので、他のすべてのノード 2 1 4 ~ 2 1 8 がそのコピーを受取るが、依然として、ブロックチェーン 2 3 0 ~ 2 3 6 のいずれのブロックにおいても記されておらず、このため、その状態は図 1 2 においてペンディング中である。なお、受取りタイミングがネットワークレイテンシまたは他の要因に依存する可能性があるので、すべてのノード 2 1 4 および 2 1 8 がトランザクションを同時に受取るとは限らないことに留意されたい。

## 【 0 1 6 6 】

なお、カタログインスタンス 2 1 4 ~ 2 1 8 がいずれも、（これらすべてが、ファイル上に検証済みデベロッパエンティティを有しているように、たとえば、ブロックチェーン 2 3 0 ~ 2 3 6 の B 1 ブロック 2 6 0 に格納されているように）トランザクション（たとえば、T 2 トランザクション）を収集、処理および伝搬し得る一方で、例示を目的として、この使用事例が第 2 のカタログインスタンス 2 1 6 によるトランザクション生成を含むことに留意されたい。

10

20

30

40

50

## 【 0 1 6 7 】

なお、いくつかの実現例においては、本来のデベロッパ登録がブロックチェーン 2 3 0 ~ 2 3 6 におけるいくつかのブロックの深さになるまで、デベロッパに対するネームスペースの登録が選択的に制限されている。この場合、この深さは、ブロックチェーン 2 3 0 ~ 2 3 6 に対する読取り / 書込みアクセスを行うカタログインスタンス 2 1 4 ~ 2 1 8 および / または他のノードに関連付けられた実現例および信頼のレベルに応じて異なってもよい。これにより、デベロッパ登録が明確に限定されていることと、我々がたとえばエコシステムにおいて用いているコンセンサスメカニズムが信頼されていないかまたは十分に信頼されていないカタログインスタンスを採用している場合にオプションの違いによってデベロッパ登録がロールバックされ得ないようにすることと、を確実にすることが容易になり得る。

10

## 【 0 1 6 8 】

図 1 3 は、第 3 のカタログインスタンス 2 1 8 が第 3 のトランザクション (すなわち、第 2 のネームスペース登録トランザクション T 3 ) 2 8 0 を開始して、これがソフトウェアコンポーネント・エコシステムにおける他のカタログインスタンス 2 1 4 , 2 1 6 のうち未検証の待ち行列 2 2 4 ~ 2 2 6 に伝搬された後における、図 1 2 の例示的なシステム 2 1 0 を示す第 7 のブロック図である。

## 【 0 1 6 9 】

T 3 トランザクション 2 8 0 はまた、トランザクションペイロードセクション 2 8 2 と、T 3 トランザクションのためのハッシュ ( T 3 ハッシュ # 4 5 b f ) を含むハッシュセクション 2 8 4 とを含む。なお、別個のハッシュセクション 2 8 4 が図 1 2 のトランザクション 2 7 0 および図 1 3 のトランザクション 2 8 0 に示されている一方で、トランザクションハッシュ ( # 4 a f 5 e 、 # 4 5 b f ) がそれぞれのトランザクション 2 7 0 および 2 8 0 を符号化するのに用いられ得ることに留意されたい。

20

## 【 0 1 7 0 】

要約すると、ブロックチェーン 2 3 0 ~ 2 3 6 に対する T 3 トランザクションのコミットの準備のために第 3 のカタログインスタンス 2 1 8 が未検証の待ち行列 2 2 4 ~ 2 2 6 に T 3 トランザクション 2 8 0 をサブミットしていることを除いては、第 3 のカタログインスタンス 2 1 8 による T 3 トランザクションの生成および伝搬は、T 2 トランザクションの生成および伝搬と同様に実行される。

30

## 【 0 1 7 1 】

図 1 4 は、ブロックチェーン 2 3 0 ~ 2 3 6 に含めるべき付加的なブロック ( 以下の図 1 5 に示されるように、B 2 ) を検証するために第 1 のカタログインスタンス 2 1 4 が選択された後における、図 1 3 の例示的なシステム 2 1 0 を示す第 8 のブロック図であって、付加的なブロックは図 1 2 の第 2 のトランザクション ( T 2 ) 2 7 0 および図 1 3 の第 3 のトランザクション ( T 3 ) 2 8 0 を組込むためのものである。なお、所与のブロックについて確認を実行するために特定のカタログインスタンス (たとえば、第 1 のカタログインスタンス 2 1 4 ) を選択することは、予め定められた選択メカニズム (たとえば、ルールおよび / またはアルゴリズム) に従って行なわれてもよいことに留意されたい。正確なルールまたはアルゴリズムは実現例特有のものであって、所与の実現例のニーズに応じて異なり得る。

40

## 【 0 1 7 2 】

第 1 のカタログインスタンス 2 1 4 は、さまざまなトランザクションハッシュ 2 7 4 および 2 8 4 (たとえば、それぞれ # 4 a f 5 e および # 4 5 b f ) を用いてツリーハッシュを計算することによって、ペンディング中のトランザクション 2 4 0 および 2 8 0 の組込みの準備のために、演算を開始する。ハッシュがどのように計算されるかについての正確な詳細は実現例特有のものであって、異なってもよい。この教示を利用できる当業者であれば、過度の実験なしに所与の実現例のニーズを満たすために、特定のハッシュアルゴリズムを容易に選択および実現し得る。

## 【 0 1 7 3 】

50

図15は、ソフトウェアコンポーネント・エコシステム210のブロックチェーン230～236に追加されるべき付加的なブロックの演算中における、第1のカタログインスタンスを示す図14の例示的なシステムを示す第9のブロック図である。B2ブロック290はトランザクションセクション292を含む。トランザクションセクション292は、組合わされたツリーハッシュ(T2トランザクションハッシュとT3トランザクションハッシュとを組合わせたものであって、T2-T3ツリーハッシュ#A324と表される)およびT2およびT3トランザクション294を含むかまたは組込んでいる。ハッシュセクション296は前のブロックハッシュ(B1ハッシュ#6ccf)および現在のブロックハッシュ(B2ハッシュ#d21d)を含む。

【0174】

したがって、第1のカタログインスタンス214は、T2(図14の70)およびT3(図14の80)のツリーハッシュを組込んだ新しいブロック290を作成する(すなわち、ブロック検証を実行する)。B1ハッシュ#6ccfは、前のハッシュ(すなわち、ブロック260のハッシュ)を表わしている。B2ハッシュは、組合わされたツリーハッシュ#A324、B1ハッシュ#6ccf、および、任意には追加情報(たとえば、制御情報および/またはトランザクションセクション292からの情報)に基づいて計算される。ブロック290が構築されて、さまざまな付随ハッシュについての計算が完了すると、B2ブロックは、図16に示されるように、新しいブロックB2として組込まれるようにブロックチェーン230～236に伝搬される。

【0175】

図16は、第1のカタログインスタンス214が付加的なブロックの処理を完了して、ソフトウェアコンポーネント・エコシステム210のブロックチェーン230～236に付加的なブロック290を伝搬した後における、図15の例示的なシステム210を示す第10のブロック図である。関連するT2(図14の70)およびT3(図14の80)トランザクション294は、B2ブロック290にコミットされた後に、未検証の待ち行列224～226から除外される。

【0176】

なお、ソフトウェアコンポーネント登録、コンポーネント使用登録、および、他のトランザクションタイプを含むさまざまなタイプのさらに別のトランザクションが3つのトランザクションT1、T2、T3について上述されたのと同様の態様で進められ得ることに留意されたい。ゼロまたはより多くのトランザクションが、(未検証の待ち行列224～226とも称される)ペンディング中のトランザクション待ち行列から選択的に処理されて確認済みブロックにされてもよく、これが、次いで、チェーン230～236に追加される。

【0177】

ブロックチェーン230～236に対して読取り/書込みアクセスを行う新しいカタログインスタンスまたは他のノードが作成されて、エコシステム210に追加される場合、新しいノードがブロックチェーン230～236を処理して、ブロックチェーン230～236における登録に基づいてシステム210についての正確なビューを再構築することができる。いくつかの実現例においては、システム210は、ブロックチェーン230～236から情報を選択的に抽出し、便宜的に異なる形式で格納するかまたは性能を向上させ得る。ブロックチェーン230～236は、ブロックチェーン230～236内の情報から再構築することができる、エコシステム210の重要な状態を記述する真実および情報のソースを表わしてもよい。

【0178】

従来より、コンピューティング・エコシステムは、概して、1.集中制御(たとえば、アップル(Apple)のアプリストア)、または、2.いわゆるワイルドウェスト(Wild West)(たとえば、独立したデベロッパ製品についてウェブをブラウズするユーザのウィンドウズ(登録商標)(Windows(登録商標))モバイルメソッド)に限定されている。

10

20

30

40

50

## 【 0 1 7 9 】

なお、1つ以上のブロックチェーン特徴を用いることは、他のタイプのデータ記録またはデータベースとは対照的に、既に信頼されているコンピューティング環境（すなわち、ネットワークノードおよびリソースのうちいくつかまたはすべてが既に制御されていて信頼されているエコシステム）においては、（以下に記載されるいくつかの実施形態の範囲内の特定の實現例の場合）実質的に不必要であり得ることに留意されたい。このようなエコシステムは、トランザクションについてのコンセンサスベースの評価を必要としない可能性もあり得るので、競合、チェイニングおよびハッシングは、ブロックチェーン以外の別のタイプの分散型データベースによって實現され得る、タイムスタンプが押された登録の分散型ストレージほど重要ではあり得ない。

10

## 【 0 1 8 0 】

したがって、（提案されたエコシステムの局面を含む）さまざまな実施形態は必ずしもすべてがブロックチェーン分散型台帳の使用を必要とするわけではない。ブロックチェーン實現例は、特に、信頼されていない環境を含む實現例における特定の實施形態を表わし得る。

## 【 0 1 8 1 】

言い換えれば、「プルーフオブワーク」と称されるメカニズムを用いる「ブロックチェーン」は、いくつかの信頼されている環境において要求される必要がない。なお、図7～図16に関連付けて上述された特定の實施形態においては、「ブロックを演算する作業」は、ハッシュの演算によって實現できることに留意されたい。

20

## 【 0 1 8 2 】

図17は、図1～図16の實施形態によって實現可能な第2の例示的な方法300のフロー図である。例示的な方法300は、ソフトウェアコンポーネント・エコシステム（たとえば、図1～図3の實施形態に従って構築されるエコシステム）を實現することを容易にする。

## 【 0 1 8 3 】

第2の例示的な方法300は第1のステップ302を含む。第1のステップ302は、ユーザ入力デバイスから1つ以上の信号を受取って、デベロッパエンティティの識別名を規定するステップを含む。たとえば、図4を参照すると、これは、デベロッパエンティティトランザクション84（すなわち、デベロッパエンティティ名86）に関するレジストラ82からの入力を受取るレジストラシステム42を含み得る。

30

## 【 0 1 8 4 】

第2のステップ304は、ユーザ入力デバイスから1つ以上の信号を受取って、（デベロッパエンティティが識別名を登録することが認可されていることを確認するために使用可能である）安全な検証メカニズムを規定するステップを含む。たとえば、図4を参照すると、セキュリティメカニズムは、デベロッパエンティティ検証のための公開鍵88およびデジタル署名90を処理するための公開鍵インフラストラクチャを實現し得る、たとえば、アイデンティティドメイン（たとえば、図2のアイデンティティドメイン44）および/またはカタログインスタンス32のセキュリティおよび許可モジュール（たとえば、図3のモジュール62）によって使用されるべき、公開鍵88およびデジタル署名90を含み得る。

40

## 【 0 1 8 5 】

第3のステップは、カタログ（たとえば、図4の第1のカタログインスタンス32）によって實現される分散型データベースにエントリを記録するステップを含む。最初の記録するステップは、ブロックチェーン20を同様に含む分散型台帳（たとえば、図3の分散型台帳14）を實現する分散型データベースの分散型未検証待ち行列38における対応するエントリを記録する前にデベロッパエンティティトランザクションデータを検証するステップを含み得る。記録されたエントリがブロックチェーン20にコミットされると、この記録されたエントリは、図3の分散型台帳14および付随のブロックチェーン20に登録されることが明らかである。最初に記録されたエントリは検証メカニズムに関連付した

50

デベロッパ名を含む。

【0186】

第4のステップ308は、カタログ（すなわち、カタログソフトウェアの実行インスタンスによって実現される分散型データベースまたは台帳）のエントリに関連付けてトランザクションを実行して、ソフトウェアコンポーネントを転送するステップを含む。このトランザクションも、登録済みのデベロッパエンティティおよび1つ以上の関連するネームスペースと関連付けてコントリビュータとして台帳に登録した登録済みコントリビュータ（たとえば、個々のデベロッパ）によるソフトウェアコンポーネントダウンロード登録トランザクションとして、分散型台帳（たとえば、分散型台帳14および付随の分散型ブロックチェーン20）に登録されてもよい。

10

【0187】

第5のステップ310は、カタログインスタンスの分散型台帳（たとえば、図3の分散型台帳14）のブロックチェーン（たとえば、図3のブロックチェーン20）における（デベロッパエンティティ登録トランザクションに対応する）カタログエントリ（すなわちエントリ）を用いて、ソフトウェアコンポーネントダウンロードトランザクションを記述する別のトランザクションの（図3の分散型台帳14のブロックチェーン20に対する）登録およびコミットを検証（すなわち、完了）するステップを含む。

【0188】

なお、説明の便宜上、カタログは、分散型台帳（たとえば、ブロックチェーン）を実現するためのコードのセットであってもよい。カタログは、利用可能なソフトウェアコンポーネントのローカルデータベースを含んでいてもよくまたは含んでいなくてもよい。なぜなら、このようなデータベースは、本教示の範囲から逸脱することなく、カタログとは別個に実現することができるからである。しかしながら、この明細書中に記載されるさまざまな実施形態においては、所与のカタログインスタンスは、利用可能なコンポーネントのローカルデータベースと対話するとともに当該ローカルデータベースを維持するためのコンピュータコード（たとえば、図3のローカルコンポーネントカタログストア36）も含む。「カタログ」および「カタログインスタンス」という語はこの明細書中では同義で用いられてもよい。

20

【0189】

なお、方法300は、本教示の範囲から逸脱することなく変更され得ることに留意されたい。たとえば、付加的なステップが追加されてもよい。いくつかのステップが置換えられてもよい。ステップは削除または交換されてもよい。さらに、追加の修正またはさらなる詳細が、方法300のさまざまなステップ320～310に追加されてもよく、および/または当該さまざまなステップ320～310から削除されてもよい。

30

【0190】

たとえば、第2の例示的な方法300はさらに、安全な検証メカニズムが公開鍵およびデジタル署名を含むことを規定している。方法300は、ソフトウェアコンポーネント・エコシステムにおける複数のノード（たとえば、カタログインスタンス）に対してカタログエントリを伝搬するステップをさらに規定し得る。

【0191】

第2の例示的な方法300は、複数のノードの各々が、分散型ブロックチェーン台帳のレプリカまたは近似レプリカを維持する他の複数のノードと協調して分散型ブロックチェーン台帳を実現するためのコードを含むカタログインスタンスを表わすことをさらに規定し得る。

40

【0192】

第2の例示的な方法300はさらに、エントリにネームスペースを割付けるための（たとえば、ネームスペース登録トランザクションに関連する分散型ブロックチェーン20に登録するための）プロセスをさらに規定し得る。

【0193】

第2の例示的な方法300はさらに、以下のステップを規定してもよく、および/また

50

は以下のステップと置換えられてもよい。以下のステップは、複数の相互通信カタログインスタンスを含む分散型ピア・ツー・ピア・コンポーネント・エコシステムのコンポーネントカタログのインスタンスを用いることで、デベロッパエンティティ名に関するデベロッパ登録エンティティ情報を受取るステップと、デベロッパエンティティ情報を検証して、デベロッパエンティティ登録の記録またはブロックとして、複数の相互通信カタログインスタンスの各々にアクセス可能であるブロックチェーンとして実現される分散型データベースにコミットすることにより、ピア・ツー・ピア・コンポーネント・エコシステムのフルノードすべてに対してデベロッパエンティティ登録情報を伝搬するステップを実行する。

#### 【0194】

第2の例示的な方法300はさらに、デベロッパエンティティ名と、登録が実行されているカタログインスタンスに関連付けられた公開鍵と、デベロッパエンティティ名に関連付けられるとともに複数の相互通信カタログインスタンスのカタログインスタンスでデベロッパエンティティアイデンティティを確認するのに使用可能な（デベロッパエンティティ機密（すなわち秘密鍵）を用いる）デジタル署名とをデベロッパエンティティによって供給するステップの後に、デベロッパ登録エンティティ情報を受取るステップが行われることを規定し得る。

#### 【0195】

ピア・ツー・ピア・コンポーネント・エコシステムのフルノードは、複数の相互通信カタログインスタンスを含み得る。第2の例示的な方法300はさらに、デベロッパエンティティ登録情報を伝搬した後、ネームスペース登録プロセス中に供給された1つ以上のネームスペースをデベロッパ登録情報に関連付けるステップによってネームスペース登録プロセスの開始および実現に応答して、デベロッパエンティティに対して1つ以上のネームスペースを割付けるステップ規定し得る。

#### 【0196】

第2の例示的な方法300はさらに、いくつかの記録またはブロックがブロックチェーンの一連の記録にコミットされる後まで、デベロッパエンティティについてのネームスペース登録プロセスの実現を選択的に遅延させるステップを含み得る。

#### 【0197】

図18は、図1～図17の実施形態を実現するのに使用可能なシステム900および付随のコンピューティング環境を示す概略ブロック図である。例示的なシステム900は、本発明の実施形態に従った分散型ソフトウェアコンポーネント・エコシステムを実現することができる。実施形態は、（たとえば、ユーザデバイスに常駐している）スタンドアロンのアプリケーションとして、または、クライアント側およびサーバ側のコードの組合せを用いて実現されるウェブベースのアプリケーションとして、実現され得る。

#### 【0198】

汎用システム900は、デスクトップコンピュータ960、ノートブックコンピュータ970、スマートフォン980、携帯電話985およびタブレット990を含むユーザデバイス960～990を含む。汎用のシステム900は、ウェブページまたは他のタイプの電子文書およびUIならびに/または実行アプリケーションを表示およびナビゲートすることができるシンクライアントコンピュータ、インターネット接続可能な携帯電話、モバイルインターネットアクセスデバイス、タブレット、電子ブック、または携帯情報端末などの如何なるタイプのユーザデバイスともインターフェイスを取ることができる。システム900は5つのユーザデバイスとともに示されてるが、ユーザデバイスはいくつでもサポートされ得る。

#### 【0199】

ウェブサーバ910は、ウェブページ、電子文書、企業データまたは他のコンテンツ、およびユーザコンピュータからの他のデータについての、ウェブブラウザおよびスタンドアロンのアプリケーションからの要求を処理するために用いられる。ウェブサーバ910はまた、企業活動に関するデータの、RSSフィードなどのプッシュデータまたは合成コ

10

20

30

40

50

ンテンツを提供してもよい。

【0200】

アプリケーションサーバ920は1つ以上のアプリケーションを操作する。アプリケーションは、任意のプログラミング言語、たとえば、Java（登録商標）、C、C++、C#など、または任意のスクリプト言語、たとえば、JavaScript（登録商標）もしくはECMAScript（European Computer Manufacturers Association Script（欧州コンピュータ製造者協会スクリプト））、Perl、PHP（Hypertext Preprocessor：ハイパーテキストプリプロセッサ）、Python、RubyまたはTCL（Tool Command Language：ツールコマンド言語）などで書込まれた1つ以上のスクリプトまたはプログラムとして実現することができる。アプリケーションは、Rails、Enterprise JavaBeans、または、.NET.などのライブラリまたはアプリケーションフレームワークを用いて構築することができる。ウェブコンテンツは、HTML（HyperText Markup Language：ハイパーテキスト・マークアップ言語）、CSS（Cascading Style Sheets：カスケーディング・スタイル・シート）、ならびに、テンプレート言語およびパーサーを含む他のウェブ技術を用いて作成され得る。

10

【0201】

アプリケーションサーバ920上で実行されるデータアプリケーションは、入力データおよびユーザコンピュータ要求を処理するように適合されており、データストレージデバイスまたはデータベース930からのデータを格納するかまたは検索することができる。データベース930は、データアプリケーションによって作成されて用いられるデータを格納している。一実施形態においては、データベース930は、SQLフォーマットコマンドまたは他のデータベースクエリ言語に回答して、データを格納、更新および検索するように適合されたリレーショナルデータベースを含む。他の実施形態は、非構造的な体系的なデータストレージアーキテクチャおよびNoSQL（Not Only SQL）データベースを用いてもよい。

20

【0202】

一実施形態においては、アプリケーションサーバ920は、プログラムまたはスクリプトを実行することができる1つ以上の汎用コンピュータを含む。一実施形態においては、ウェブサーバ910は、1つ以上の汎用コンピュータ上で実行されるアプリケーションとして実現される。ウェブサーバ910およびアプリケーションサーバ920は同じコンピュータ上で組み合わせられて実行されてもよい。

30

【0203】

電子通信ネットワーク940～950は、ユーザコンピュータ960～990と、ウェブサーバ910と、アプリケーションサーバ920と、データベース930との間の通信を可能にする。一実施形態においては、ネットワーク940～950はさらに、有線ネットワーク940および無線ネットワーク950を含む電気通信デバイスまたは光通信デバイスのいずれかの形態を含み得る。ネットワーク940～950はまた、イーサネット（登録商標）ネットワークなどの1つ以上のローカルエリアネットワーク、インターネットなどのワイドエリアネットワーク、セルラーキャリアデータネットワーク；および、仮想プライベートネットワークなどの仮想ネットワークを組み込んでいてもよい。

40

【0204】

当該システムは、本発明の一実施形態に従ってアプリケーションを実行するための一例である。別の実施形態においては、アプリケーションサーバ910、ウェブサーバ920および任意にはデータベース930は、単一のサーバコンピュータアプリケーションおよびシステムと組み合わせることができる。さらに別の実施形態においては、アプリケーションサーバ910、ウェブサーバ920およびデータベース930の1つ以上を実現するために、仮想化および仮想マシンアプリケーションが用いられてもよい。

【0205】

さらに別の実施形態においては、ウェブおよびアプリケーションサービング機能のすべてまたは一部は、ユーザコンピュータの各々の上で実行されるアプリケーションに一体化されていてもよい。たとえば、ユーザコンピュータ上のJavaScriptアプリケーションは、

50

アプリケーションのデータ部分および表示部分を検索するかまたは分析するために用いられてもよい。

【0206】

図2および図18に関連付けて、一例として、ウェブサーバ910、アプリケーションサーバ920およびデータストレージデバイス/データベース930は、ブラウザを介して個々のコンピュータシステムにさらにアクセス可能であるカタログインスタンス32に対応するサーバ側アプリケーションをホストすることによって、図2の分散型台帳14を実現するために用いられてもよい。代替的には、または付加的には、個々のコンピューティングデバイス950、985、970、980、990は、有線ネットワーク940および/または無線ネットワーク950を用いてピア・ツー・ピア・コンポーネント・エコシステムにネットワーク化されるカタログインスタンスソフトウェアを実行してもよい。

10

【0207】

図19は、図1～図18の実施形態を実現するために使用可能なコンピューティングデバイスを示す概略ブロック図である。図19のシステム500は、この明細書中における特定の実現例に記載されているステップを実行することを容易にするものとして記載されている一方で、記載されたステップを実行するために、システム500の任意の好適なコンポーネントもしくはコンポーネントの組合せ、または、システム500に関連付けられた任意の好適な一つのプロセッサまたは複数のプロセッサが用いられてもよい。

【0208】

図19は、この明細書中に記載される実現例のために用いられ得る例示的なコンピューティングシステム500のブロック図を示す。たとえば、コンピューティングシステム500は、図18のサーバデバイス910および920を実現するとともに、この明細書中に記載される方法の実現例を実行するために用いられてもよい。いくつかの実現例においては、コンピューティングシステム500は、プロセッサ502、オペレーティングシステム504、メモリ506、および入出力(input/output: I/O)インターフェイス508を含んでいてもよい。さまざまな実現例においては、プロセッサ502は、この明細書中に記載されるさまざまな機能および特徴を実現するとともに、この明細書中に記載される方法の実現例を実行するために用いられてもよい。プロセッサ502がこの明細書中に記載される実現例を実行するものとして記載されているが、システム500の任意の好適なコンポーネントもしくはコンポーネントの組合せ、または、システム500に関連付けられた任意の好適な1つのプロセッサまたは複数のプロセッサ、または、任意の好適なシステムが、記載されるステップを実行してもよい。この明細書中に記載される実現例は、ユーザデバイス上で、サーバ上で、またはこれら両方の組合せに対して実行されてもよい。

20

30

【0209】

コンピューティングデバイス500はまた、メモリ506上に格納され得るか、または、他の好適な任意の記憶位置もしくはコンピュータ読取可能媒体上に格納され得るソフトウェアアプリケーション510を含む。ソフトウェアアプリケーション510は、プロセッサ502がこの明細書中に記載される機能および他の機能を実行することを可能にする命令を提供する。コンピューティングシステム500のコンポーネントは、1つ以上のプロセッサによって、または、ハードウェアデバイスの任意の組合せによって、さらには、ハードウェア、ソフトウェア、ファームウェアなどの任意の組合せによって、実現されてもよい。

40

【0210】

例示を容易にするために、図19は、プロセッサ502、オペレーティングシステム504、メモリ506、I/Oインターフェイス508およびソフトウェアアプリケーション510の各々ごとに1つのブロックを示す。これらのブロック502、504、506、508および510は、複数のプロセッサ、オペレーティングシステム、メモリ、I/Oインターフェイス、およびソフトウェアアプリケーションを表わし得る。さまざまな実現例においては、コンピューティングシステム500は図示されるコンポーネントをすべ

50

て有していなくてもよく、および/または、この明細書中に示される以外の他のタイプのコンポーネント、もしくはこの明細書中に記載されるものに加えて他のタイプのコンポーネントを含む他の要素を有していてもよい。

【0211】

一例として、図3および図19を参照すると、図19のコンピューティングデバイス500は図3の個々のデベロッパシステム62を実現するために用いられてもよい。

【0212】

特定の実施形態に関して説明がなされてきたが、これらの特定の実施形態は単に例示的なものに過ぎず、限定的なものではない。

【0213】

特定の実施形態に関して説明がなされてきたが、これらの特定の実施形態は単に例示的なものに過ぎず、限定的なものではない。

【0214】

如何なる好適なプログラミング言語を用いても、C、C++、Java、アセンブリ言語などを含む特定の実施形態のルーチンを実現することができる。手続き指向型またはオブジェクト指向型などのさまざまなプログラミング技術を用いることができる。ルーチンは単一の処理装置または複数のプロセッサ上で実行することができる。ステップ、動作または演算は特定の順序で表わされ得るが、この順序はさまざまな特定の実施形態において変更されてもよい。いくつかの特定の実施形態においては、この明細書中で連続的なものとして示されている複数のステップは同時に実行することができる。

【0215】

特定の実施形態は、命令実行システム、装置、システムもしくはデバイスによって、またはそれらに関連付けて用いられるようにコンピュータ読取可能記憶媒体において実現されてもよい。特定の実施形態は、ソフトウェアもしくはハードウェアまたはそれらの組合せで制御論理の形態で実現することができる。制御論理は、1つ以上のプロセッサによって実行されると、特定の実施形態において記載されているものを実行するように動作可能であり得る。

【0216】

特定の実施形態は、特定用途向け集積回路、プログラマブルロジックデバイス、フィールドプログラマブルゲートアレイ、光学的、化学的、生物学的、量子的またはナノ設計型のシステム、コンポーネントおよびメカニズムを用いることによる、プログラムド汎用デジタルコンピュータを用いることによって実現されてもよい。一般に、特定の実施形態の機能は当該技術において公知である如何なる手段によっても達成することができる。分散型のネットワーク化システム、コンポーネントおよび/または回路を用いることができる。データの通信または転送は有線、無線または他のいずれかの手段によってなされてもよい。

【0217】

添付の図面/図に示される要素のうちの一つ以上が、特定用途に応じて有用となるように、さらに分離された態様もしくは一体化された態様で、または場合によっては削除されるかもしくは動作不能にされた態様で実現可能であることも認識されるだろう。コンピュータが上述の方法のうちの一つ以上を実行することを可能にするために機械読取り可能媒体に記憶することができるプログラムまたはコードを実現することも精神および範囲内にある。

【0218】

この明細書中で、かつ添付の特許請求の範囲全体にわたって用いられる「a」、「an」および「the」という語は、文脈に別段明確な指示がない限り、複数の指示対象を含んでいる。また、この明細書中で、かつ添付の特許請求の範囲全体にわたって用いられる「in」の意味は、文脈に別段明確な指示がない限り、「において(in)」および「の上に(on)」の意味を含んでいる。

【0219】

10

20

30

40

50

このように、特定の実施形態をこの明細書中に記載してきたが、上述の開示において変更、さまざまな変形および置換えの許容範囲が意図されており、場合によっては、特定の実施形態のいくつかの特徴が、上述の範囲および精神から逸脱することなく他の特徴を対応させて使用することなく採用され得ることが認識されるだろう。したがって、特定の状況または材料を本質的な範囲および精神に適合させるために多くの変更がなされてもよい。

【 図 1 】

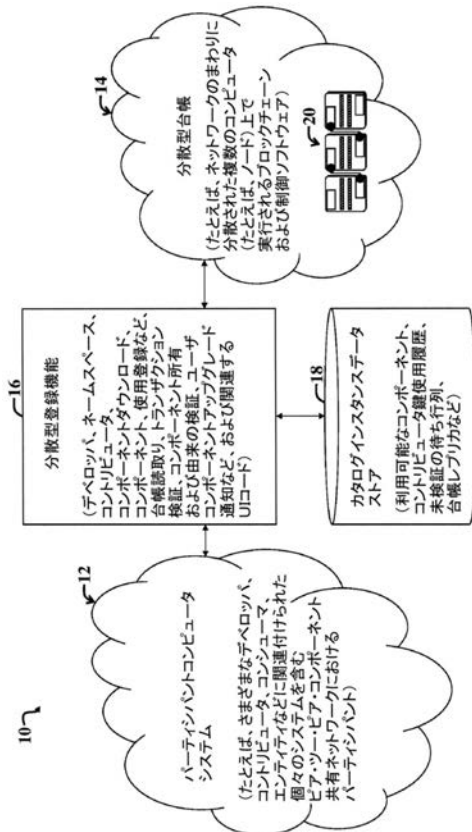


FIG. 1

【 図 2 】

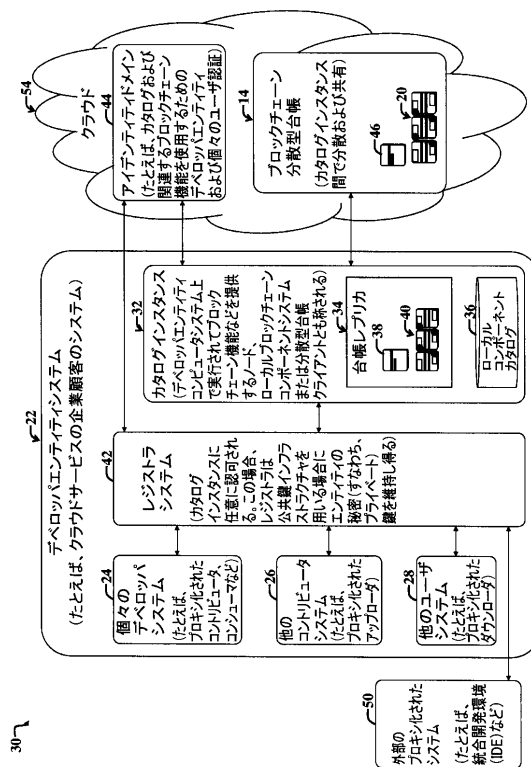


FIG. 2

【 図 3 】

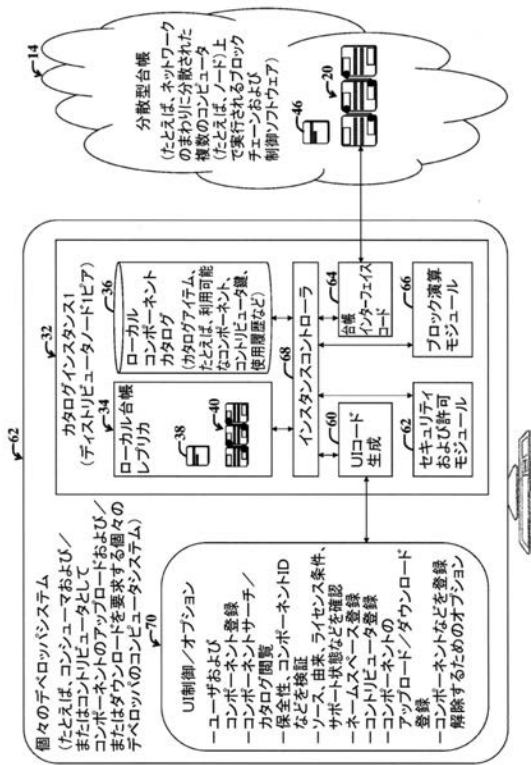


FIG. 3

【 図 4 】

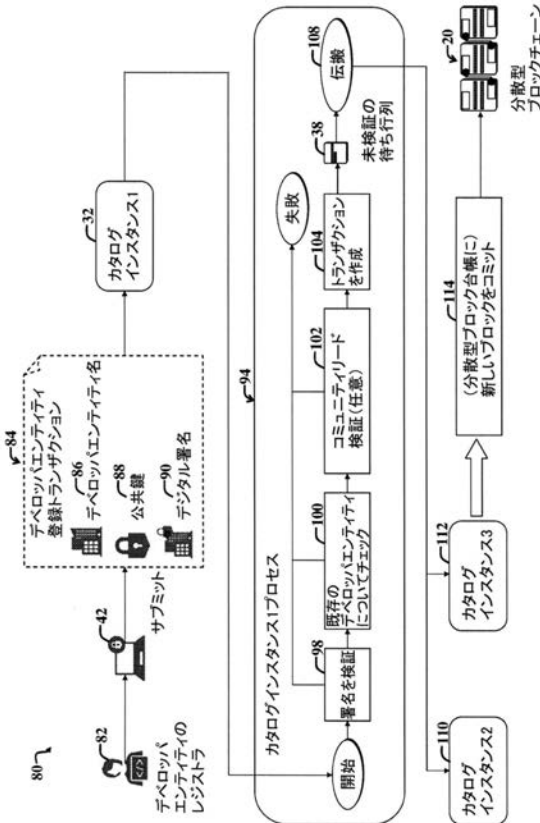


FIG. 4

【 図 5 】

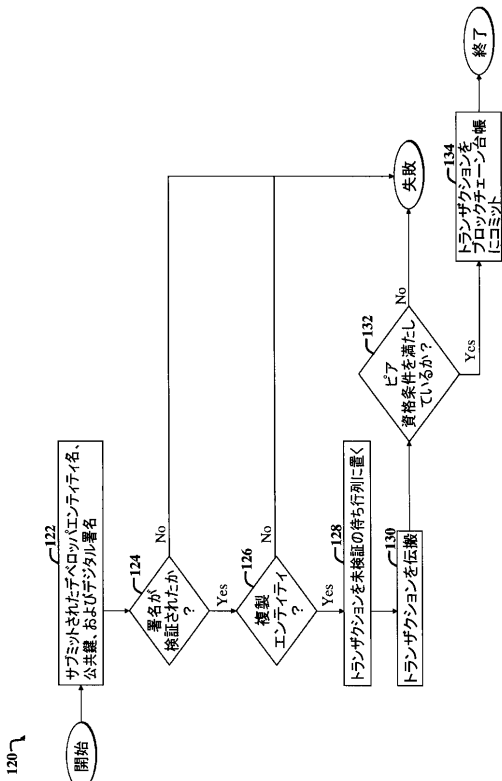


FIG. 5

【 図 6 】

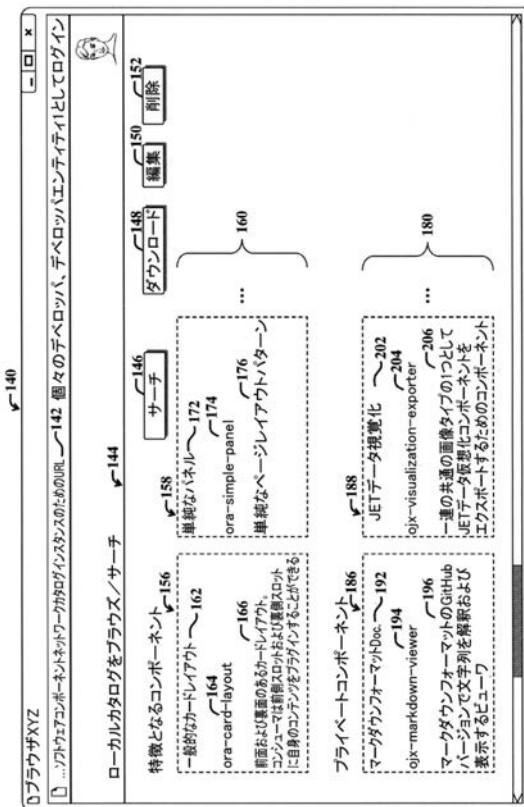


FIG. 6

【 図 7 】

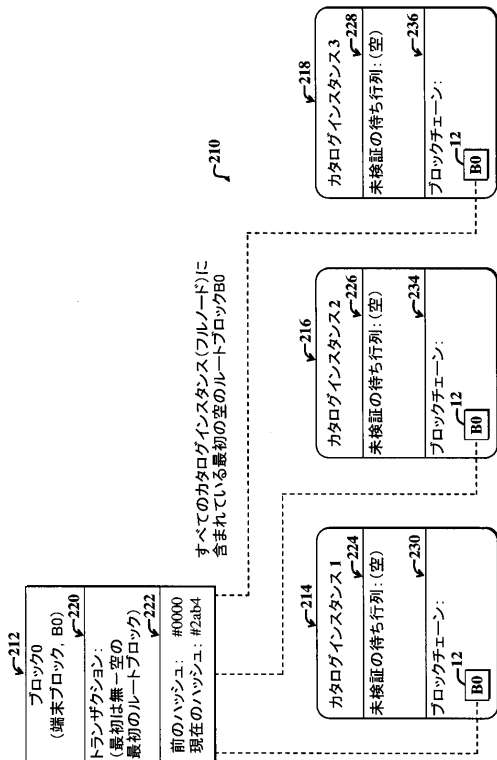


FIG. 7

【 図 8 】

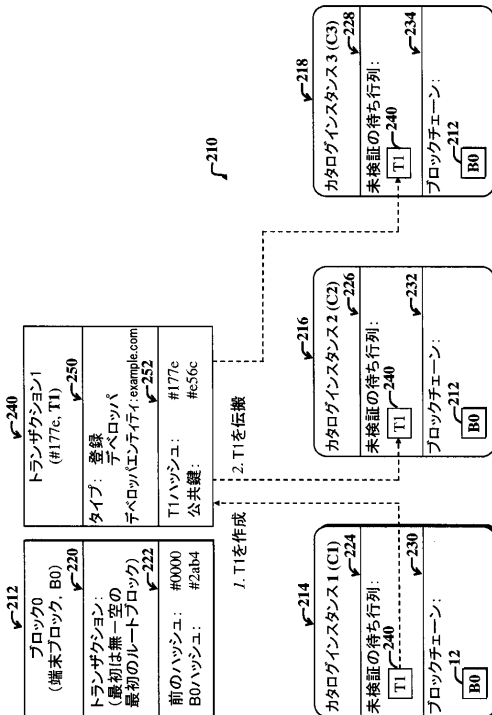


FIG. 8

【 図 9 】

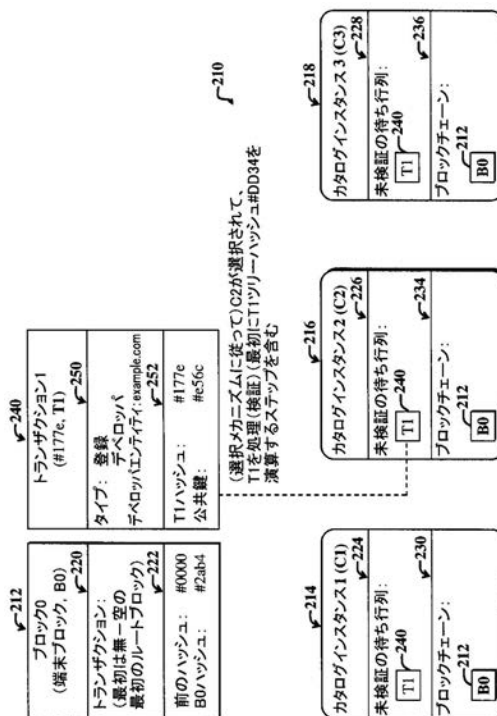


FIG. 9

【 図 10 】

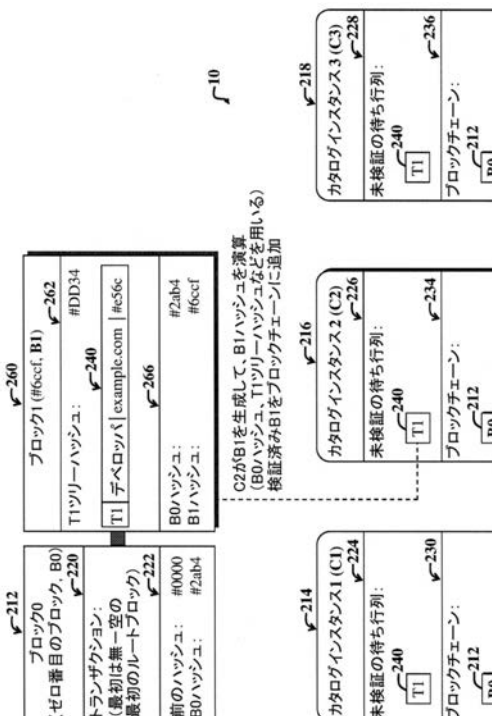


FIG. 10

【 図 1 1 】

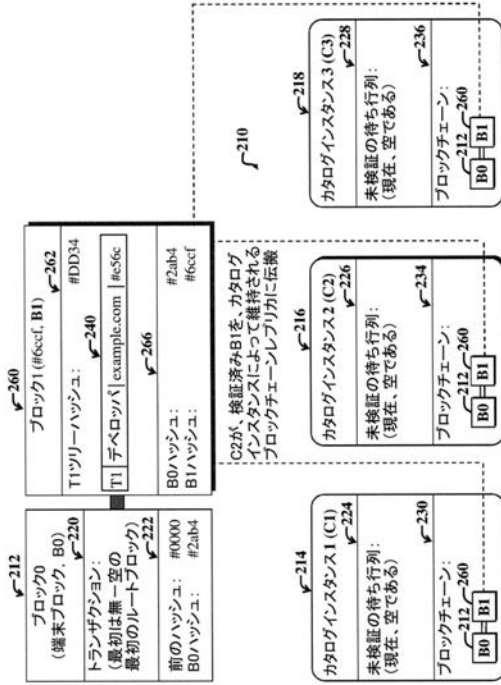


FIG. 11

【 図 1 2 】

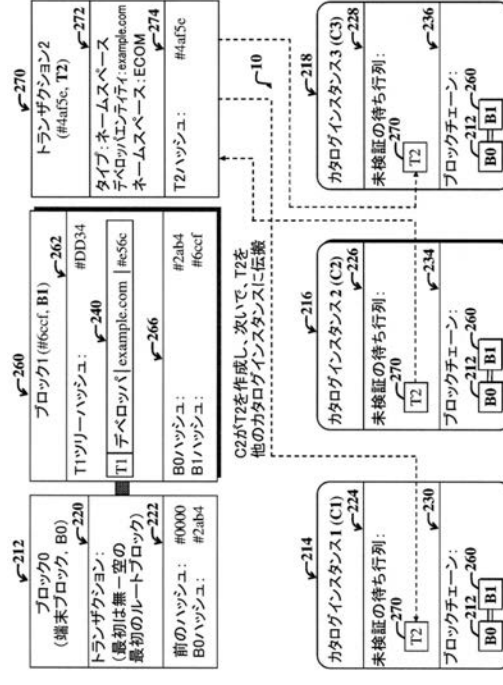


FIG. 12

【 図 1 3 】

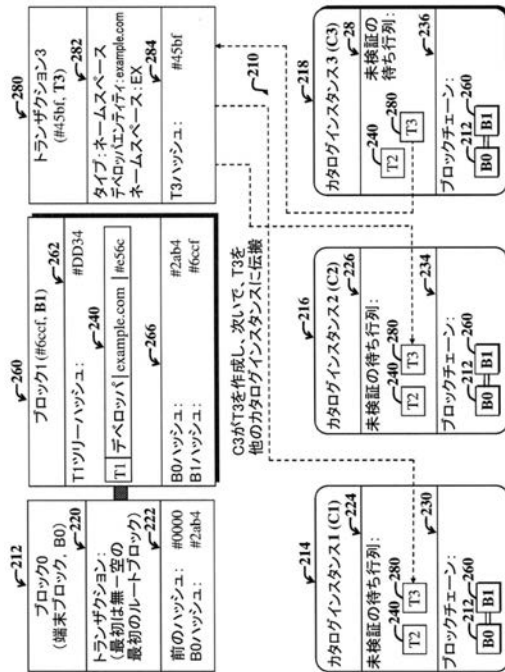


FIG. 13

【 図 1 4 】

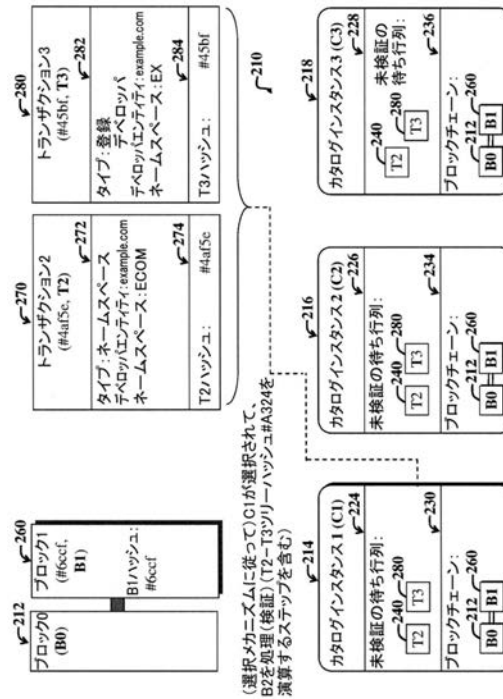


FIG. 14

【 図 1 5 】

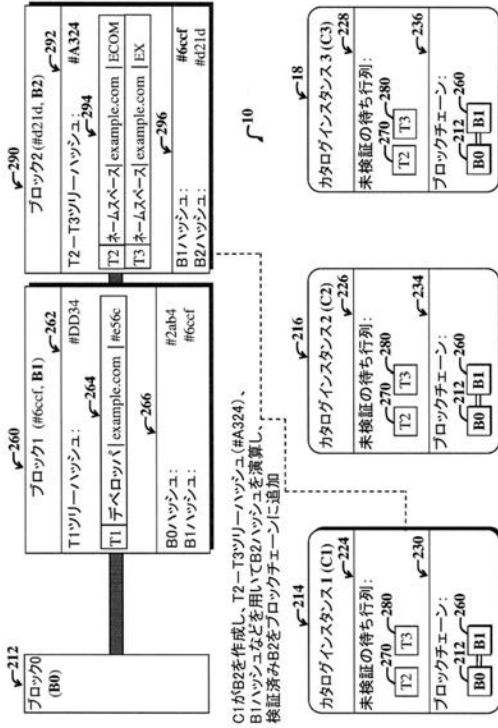


FIG. 15

【 図 1 6 】

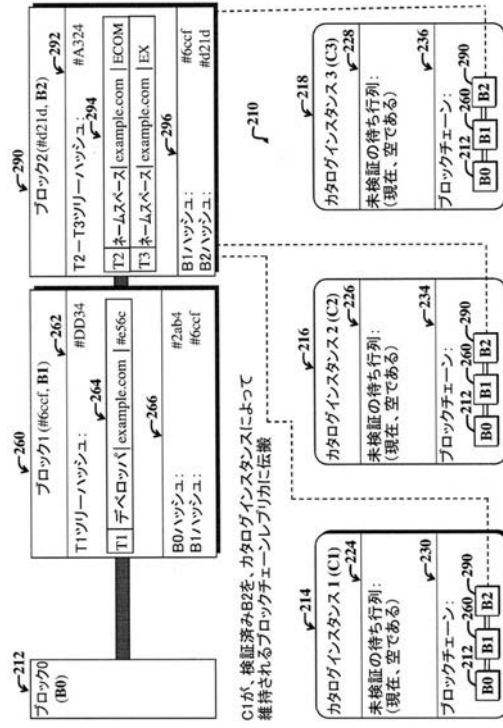


FIG. 16

【 図 1 7 】

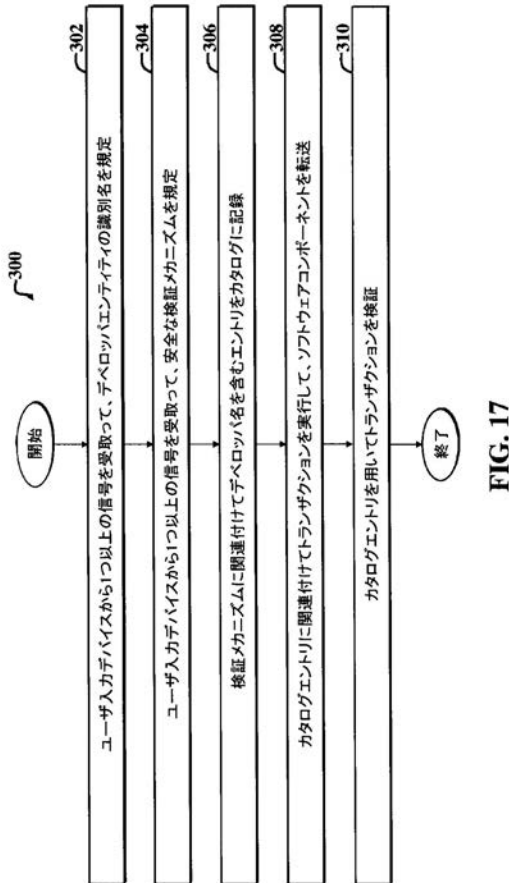


FIG. 17

【 図 1 8 】

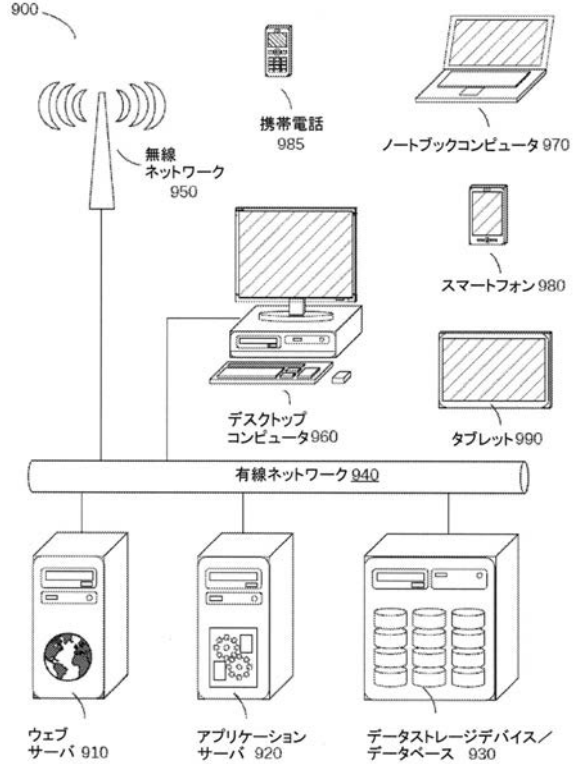


FIG. 18

【 図 1 9 】

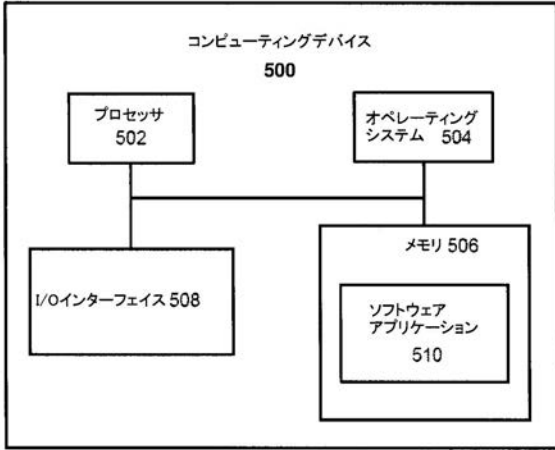


FIG. 19

## 【 国際調査報告 】

## INTERNATIONAL SEARCH REPORT

International application No  
PCT/US2018/040105

A. CLASSIFICATION OF SUBJECT MATTER INV. H04L9/32 H04L9/12 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04L G06F H04W		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EP0-Internal, WPI Data, INSPEC		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	FOTIOU NIKOS ET AL: "Decentralized name-based security for content distribution using blockchains", 2016 IEEE CONFERENCE ON COMPUTER COMMUNICATIONS WORKSHOPS (INFOCOM WKSHPs), IEEE, 10 April 2016 (2016-04-10), pages 415-420, XP032957328, DOI: 10.1109/INFOCOMW.2016.7562112 [retrieved on 2016-09-06] the whole document ----- -/--	1-21
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> See patent family annex.
* Special categories of cited documents :		
<p>"A" document defining the general state of the art which is not considered to be of particular relevance</p> <p>"E" earlier application or patent but published on or after the international filing date</p> <p>"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)</p> <p>"O" document referring to an oral disclosure, use, exhibition or other means</p> <p>"P" document published prior to the international filing date but later than the priority date claimed</p>		<p>"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</p> <p>"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</p> <p>"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</p> <p>"&amp;" document member of the same patent family</p>
Date of the actual completion of the international search 25 September 2018		Date of mailing of the international search report 02/10/2018
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel (+31-70) 340-2040, Fax (+31-70) 340-3016		Authorized officer Mariggis, Athanasios

2

## INTERNATIONAL SEARCH REPORT

International application No PCT/US2018/040105
---

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>Ori Jacobovitz: "Blockchain for Identity Management",            1 December 2016 (2016-12-01), XP055503803,            Retrieved from the Internet:            URL:https://www.cs.bgu.ac.il/~frankel/TechnicalReports/2016/16-02.pdf            [retrieved on 2018-08-31]            the whole document</p>	1-21
X	<p>-----            WO 2017/079795 A1 (HANNA ROGER [AU])            18 May 2017 (2017-05-18)            abstract            paragraphs [0001] - [0088]            figure 1            -----</p>	1-21

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No

PCT/US2018/040105

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2017079795 A1	18-05-2017	AU 2016354668 A1	28-06-2018
		EP 3374952 A1	19-09-2018
		WO 2017079795 A1	18-05-2017
-----			

## フロントページの続き

(31)優先権主張番号 62/553,406

(32)優先日 平成29年9月1日(2017.9.1)

(33)優先権主張国・地域又は機関  
米国(US)

(81)指定国・地域 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, 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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, 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