(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2020-145647
(P2020-145647A)
(43)公開日　令和2年9月10日(2020.9.10)

| (51)Int.Cl. | F I | テーマコード（参考） |
|---|---|---|
| H04L 12/825 (2013.01) | H04L 12/825 | 5K030 |

審査請求 未請求　請求項の数 12　ＯＬ　（全 25 頁）

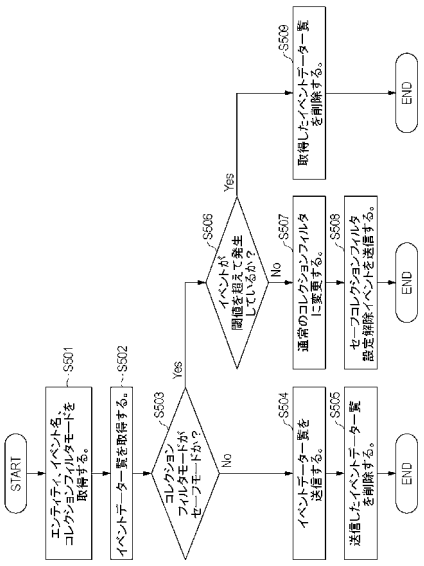
(21)出願番号　特願2019-42587(P2019-42587)
(22)出願日　平成31年3月8日(2019.3.8)

(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　100126240
弁理士　阿部　琢磨
(74)代理人　100124442
弁理士　黒岩　創吾
(72)発明者　結城　直人
東京都大田区下丸子３丁目３０番２号キヤノン株式会社内

Ｆターム(参考)　5K030 GA13　HA08　HC01　LC01　LC11　MB09

(54)【発明の名称】システム、クライアント端末、制御方法、および、プログラム

(57)【要約】
【課題】　所定のクライアントから大量のデータが送信された場合に、受信サービスにおいて他のクライアントのデータ処理を遅延させないための仕組みを提供する
【解決手段】　受信サービスシステムは、所定のクライアント端末から単位時間あたりに所定値を超えるデータ量が送信された場合に、第１の送信ルールを所定のクライアント端末に対して送信する。複数のクライアント端末のそれぞれは、受信サービスシステムに対して、クライアント端末で発生したイベントに関するデータを送信し、受信サービスシステムが送信した第１の送信ルールを保持する。クライアント端末は、第１の送信ルールに従い、単位時間あたりに送信するデータ量が所定値以下となるように、クライアント端末で発生したイベントに関するデータのうち少なくとも一部のデータを受信サービスシステムに対して送信しない。
【選択図】　図５



【特許請求の範囲】
【請求項１】
複数のクライアント端末と、前記複数のクライアント端末からデータを受信する受信サービスシステムと、を含むシステムであって、
前記受信サービスシステムは、
前記複数のクライアント端末からデータを受信する受信手段と、
前記複数のクライアント端末のうちの所定のクライアント端末から単位時間あたりに所定値を超えるデータ量が送信された場合に、第１の送信ルールを前記所定のクライアント端末に対して送信する第１の送信手段と、を有し、
前記複数のクライアント端末のそれぞれは、
前記受信手段に対して、前記クライアント端末で発生したイベントに関するデータを送信する第２の送信手段と、
前記第１の送信手段が送信した前記第１の送信ルールを保持する保持手段と、を有し、
前記第２の送信手段は、前記保持される前記第１の送信ルールに従い、単位時間あたりに送信するデータ量が所定値以下となるように、前記クライアント端末で発生したイベントに関するデータのうち少なくとも一部のデータを前記受信手段に対して送信しないことを特徴とするシステム。
【請求項２】
前記第２の送信手段は、前記保持される前記第１の送信ルールに従い、前記クライアント端末で発生したイベントに関するデータのいずれも前記受信手段に対して送信しないことを特徴とする請求項１に記載のシステム。
【請求項３】
前記複数のクライアント端末のそれぞれは、
前記クライアント端末で発生したイベントに関するデータの量が、前記受信手段が単位時間あたりに受信可能なデータ量を示す所定値を超えるか否かを判定する判定手段と、
前記クライアント端末で発生したイベントに関するデータの量が前記所定値以下であると判定された場合、前記受信サービスシステムに対して第１の通知を行う第１の通知手段と、を更に有し、
前記第１の送信手段は、前記第１の通知を受信した場合に、前記第１の送信ルールとは別の第２の送信ルールを前記所定のクライアント端末に対して送信し、
前記保持手段は、前記第１の送信手段が送信した前記第２の送信ルールを保持し、
前記第２の送信手段は、前記保持手段で保持される前記第１の送信ルールが前記第２の送信ルールに変更された場合に、前記第２の送信ルールに従い、前記クライアント端末で発生したイベントに関するデータのいずれも前記受信手段に対して送信することを特徴とする請求項１または２に記載のシステム。
【請求項４】
前記システムには、前記受信サービスシステムから受け取ったクライアント端末のデータを処理するリソースサービスシステムが更に含まれ、
前記受信サービスシステムは、
前記第１の送信手段が第１の送信ルールを前記所定のクライアント端末に対して送信した場合に、前記第１の送信手段が第１の送信ルールを前記所定のクライアント端末に対して送信したことを示す第１のイベントを、前記リソースサービスシステムに対して通知する第２の通知手段を更に有することを特徴とする請求項１乃至３のいずれか１項に記載のシステム。
【請求項５】
前記第２の送信手段は、前記保持手段で保持される前記第１の送信ルールが前記第２の送信ルールに変更された場合に、前記保持手段で保持される前記第１の送信ルールが前記第２の送信ルールに変更されたことを示す第２のイベントを前記受信手段に対して送信し、
前記第２の通知手段は、前記受信手段が前記第２のイベントを受信した場合、前記リソ

ースサービスシステムに対して前記第２のイベントを通知することを特徴とする請求項４に記載のシステム。
【請求項６】
前記システムには、クライアント端末の認証情報を管理する認証管理サーバが更に含まれ、
前記受信サービスシステムは、
前記第１の送信手段が第１の送信ルールを前記所定のクライアント端末に対して送信してから一定時間が経過した場合であって、かつ、前記受信手段が前記所定のクライアント端末から受信したデータに前記第１の送信ルールを示す情報が含まれない際には、前記所定のクライアント端末の認証情報を削除するよう前記認証管理サーバに対して要求する要求手段を更に有することを特徴とする請求項１乃至５のいずれか１項に記載のシステム。
【請求項７】
前記システムには、前記複数のクライアント端末それぞれの状態情報を管理する状態管理サーバが更に含まれ、
前記第１の送信手段は、所定のクライアント端末から単位時間あたりに所定値を超えるデータ量が送信された場合に、前記状態管理サーバで管理される前記所定のクライアント端末の状態情報を変更することで、前記第１の送信ルールを前記所定のクライアント端末に対して送信することを特徴とする請求項１乃至６のいずれか１項に記載のシステム。
【請求項８】
クライアント端末からデータを受信する受信サービスシステムと通信可能なクライアント端末であって、
前記クライアント端末で発生したイベントに関するデータを前記受信サービスシステムに送信する送信手段と、を有し、
前記送信手段は、前記クライアント端末で発生したイベントに関するデータの量が、前記受信サービスシステムが単位時間あたりに受信可能なデータ量を示す所定値を超える場合、単位時間あたりに送信するデータ量が前記所定値以下となるように、前記データの少なくとも一部を前記受信サービスシステムに対して送信しないことを特徴とするクライアント端末。
【請求項９】
前記クライアント端末で発生したイベントに関するデータの量が、前記受信サービスシステムが単位時間あたりに受信可能なデータ量を示す所定値を超えるか否かを判定する判定手段と、
イベントデータを前記受信サービスシステムに送信する際に用いる送信ルールを保持する保持手段と、
前記クライアント端末で発生したイベントに関するデータの量が、前記受信サービスシステムが単位時間あたりに受信可能なデータ量を示す所定値を超える場合、前記保持手段で保持される送信ルールを変更する変更手段と、を更に有し、
前記送信手段は、前記変更された送信ルールに従い、前記データのいずれも前記受信サービスシステムに対して送信しないことを特徴とする請求項８に記載のクライアント端末。
【請求項１０】
複数のクライアント端末と、前記複数のクライアント端末からデータを受信する受信サービスシステムと、を含むシステムの制御方法であって、
前記受信サービスシステムが、前記複数のクライアント端末からデータを受信する第１の受信工程と、
前記受信サービスシステムが、前記複数のクライアント端末のうちの所定のクライアント端末から単位時間あたりに所定値を超えるデータ量が送信された場合に、第１の送信ルールを前記所定のクライアント端末に対して送信する第１の送信工程と、を有し、
前記複数のクライアント端末のそれぞれが、前記受信サービスシステムに対して、前記クライアント端末で発生したイベントに関するデータを送信する第２の送信工程と、

前記複数のクライアント端末のそれぞれが、前記第１の送信工程で送信された前記第１の送信ルールを保持する保持工程と、を有し、
前記第２の送信工程では、前記保持される前記第１の送信ルールに従い、単位時間あたりに送信するデータ量が所定値以下となるように、前記クライアント端末で発生したイベントに関するデータのうち少なくとも一部のデータを前記受信サービスシステムに対して送信しないことを特徴とする制御方法。
【請求項１１】
クライアント端末からデータを受信する受信サービスシステムと通信可能なクライアント端末の制御方法であって、
前記クライアント端末で発生したイベントに関するデータを前記受信サービスシステムに送信する送信工程と、を有し、
前記送信工程では、前記クライアント端末で発生したイベントに関するデータの量が、前記受信サービスシステムが単位時間あたりに受信可能なデータ量を示す所定値を超える場合、単位時間あたりに送信するデータ量が前記所定値以下となるように、前記データの少なくとも一部を前記受信サービスシステムに対して送信しないことを特徴とする制御方法。
【請求項１２】
請求項８または９に記載の手段としてコンピューターを機能させるためのプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、複数のクライアント端末と受信サービスシステムとを含むシステム、クライアント端末、制御方法、および、プログラムに関する。
【背景技術】
【０００２】
近年、インターネットに家電製品や自動車などをクライアント端末として接続する「Ｉｎｔｅｒｎｅｔ　ｏｆ　Ｔｈｉｎｇｓ」（以降、ＩｏＴと称する）を実現するシステムが利用されている。また、画像形成装置の多機能化が進み、Ｍｕｌｔｉ　Ｆｕｎｃｔｉｏｎ　Ｐｒｉｎｔｅｒ（以降、ＭＦＰと称する）と呼ばれる複合機もＩｏＴクライアントとして対応することが可能となった。
【０００３】
ＩｏＴシステムは、ＩｏＴクライアントに接続されたセンサー情報など膨大な量のデータを収集、分析することで、ＩｏＴクライアント（単にクライアントとも称する）及びそれを利用するユーザに対してサービス、付加価値を提供する。ＩｏＴシステムにおいては、前述した膨大な量のデータを、同じく膨大な数のクライアントから受信するようになってきている。
【０００４】
このような膨大なデータをリアルタイムに受信する処理では、刻一刻と変動するトラフィックの監視が重要である。トラフィックのデータを受信するサーバは、ＡＰＩを備えた従来型のリソースサーバーが単独で対応することは困難である。そこで、データの収集には、データ受信専用のサービス（受信サービス）を利用し、サービスを提供する処理（リソースサービス）は非同期処理として分離して対応することが一般的になりつつある。
【０００５】
ＩｏＴクライアントは受信サービスにデータを送信し、受信サービスはデータをバッファリングし、リソースサービスからのデータ受信要求に対してバッファリングからのデータを供給する。このようにして、リソースサービスはＩｏＴクライアントからの変動する膨大なトラフィックを意識することなく、データの処理に専念することができる。
【０００６】
また、ストリーム受信サービスからデータを転送されるリソースサービスは、複数存在することが可能である。リソースサービスはストリーム受信サービスとは非同期で処理を

実行しているため、ＩｏＴシステムの開発において、提供するサービスの種類ごとに柔軟にリソースサービスを追加することができる。

【０００７】

ここで特許文献１には、監視対象物のセンサが出力するセンサデータをゲートウェイから受信するサーバがセンサデータに基づく処理を行うことで、監視対象物の故障予測を行うデータ収集システムについて記載されている。サーバは、監視対象物に異常または異常の予兆が発生したことを検出した場合、収集するセンサデータの種類や収集頻度を増やすための収集ルールをゲートウェイに送信することが記載されている。

【先行技術文献】

【特許文献】

【０００８】

【特許文献１】特開２０１６－１６３２４２号公報

【発明の概要】

【発明が解決しようとする課題】

【０００９】

例えばハードディスク故障などを起こしたクライアントが、異常なデータを通常よりも高頻度で大量に受信サービスに送信する場合がある。しかしながら、データの処理を実行する受信サービス内のリソースの量には制限がある。そのため、一部のクライアントが通常よりも高頻度で大量のデータを送信すると、受信サービスが処理すべきデータ量が増え、結果として、受信サービスにおいて他の正常なクライアントからのデータの処理に遅延が生じてしまう。

【００１０】

そこで本発明は、所定のクライアントから大量のデータが送信された場合に、受信サービスにおいて他のクライアントのデータ処理を遅延させないための仕組みを提供することを目的とする。

【課題を解決するための手段】

【００１１】

上記課題を解決するために、本発明は、複数のクライアント端末と、前記複数のクライアント端末からデータを受信する受信サービスシステムと、を含むシステムであって、前記受信サービスシステムは、前記複数のクライアント端末からデータを受信する受信手段と、前記複数のクライアント端末のうちの所定のクライアント端末から単位時間あたりに所定値を超えるデータ量が送信された場合に、第１の送信ルールを前記所定のクライアント端末に対して送信する第１の送信手段と、を有し、前記複数のクライアント端末のそれぞれは、前記受信手段に対して、前記クライアント端末で発生したイベントに関するデータを送信する第２の送信手段と、前記第１の送信手段が送信した前記第１の送信ルールを保持する保持手段と、を有し、前記第２の送信手段は、前記保持される前記第１の送信ルールに従い、単位時間あたりに送信するデータ量が所定値以下となるように、前記クライアント端末で発生したイベントに関するデータのうち少なくとも一部のデータを前記受信手段に対して送信しないことを特徴とする。

【発明の効果】

【００１２】

本発明により、所定のクライアントから大量のデータが送信された場合に、受信サービスにおいて他のクライアントのデータ処理が遅延することを回避できる。

【図面の簡単な説明】

【００１３】

【図１】システム全体図

【図２】ハードウェア構成図

【図３】ソフトウェア構成図

【図４】クライアント端末がストリーム受信サーバにデータを送信するまでの処理のシーケンス

【図５】クライアント端末がストリーム受信サーバにデータを送信する処理のフローチャート
【図６】コレクションフィルタの例を示す図
【図７】イベントデータの例を示す図
【図８】ストリーム処理モジュールの処理のシーケンス
【図９】ストリーム処理モジュールの処理のフローチャート
【図１０】第２の実施の形態におけるストリーム処理モジュールの処理のフローチャート
【図１１】第３の実施の形態におけるクライアント端末がストリーム受信サーバにデータを送信する処理のフローチャート
【図１２】第３の実施の形態におけるストリーム処理モジュールの処理のフローチャート
【発明を実施するための形態】
【００１４】
以下、本発明を実施するための形態について図面を用いて説明する。
【００１５】
（実施例１）
本実施例においては、ネットワーク上の各サーバにアプリケーションが設置されていることとする。また、各アプリケーションはクライアント端末と連携し、様々な機能を提供することとする。このような機能を提供する実体をサービスと呼称し、機能をクライアント端末に提供することをサービスの提供と呼称する。また、複数のサーバと複数のアプリケーションから複数のサービスを連携しあい提供するサービスを統合サービスと呼称する。
【００１６】
図１は、本実施例に係る情報処理システムであるデータ収集基盤システムの全体構成を示す図である。本実施例に係る情報処理システムは、クライアント端末のデータを受信する受信サービスシステム、および、受信したデータを処理してサービスを提供する複数のリソースサービスシステムを含む。受信サービスシステムと各リソースサービスシステムとは、非同期でデータ処理を行っている。本実施例に係る情報処理システムで提供されるサービスは、例えば、クライアント端末のデータをバックアップするサービスや、クライアント端末に接続されたセンサー情報から稼働状況を分析するサービスなどがある。
【００１７】
ネットワーク１００は本システムの各構成要素を通信可能に接続するＷｉｄｅ　Ａｒｅａ　Ｎｅｔｗｏｒｋ（以降、ＷＡＮと称する）である。
【００１８】
ネットワーク１０１、１１１、１２１は本システムの各構成要素を通信可能に接続するＬｏｃａｌ　Ａｒｅａ　Ｎｅｔｗｏｒｋ（以降、ＬＡＮと称する）である。
【００１９】
クライアント端末１０２は、サービスを利用するためのパソコン、モバイル端末、画像形成装置などの機器である。
【００２０】
第１の認証認可サーバ１１２は、クライアント端末１０２がサービスを利用するために必要となる認証・認可を実現するための認証管理サーバである。第１の認証認可サーバ１１２は、クライアント端末１０２のデータ収集サーバ１１３、データ利用サーバ１１４へのアクセスを制御する。
【００２１】
データ収集サーバ１１３は、クライアント端末１０２のデータを収集するサービスと後述のデータ利用サーバ１１４へ収集したデータを通知するサービスを提供するサーバである。データの収集には後述のストリーム受信サーバ１２３、データの通知には後述のメッセージサーバ１２４を利用する。
【００２２】
データ利用サーバ１１４は、データ収集サーバ１１３で収集したデータを利用したサー

ビスを提供するサーバである。例えば、データ利用サーバ１１４はクライアント端末１０２のデータをバックアップするサービスや、クライアント端末１０２に接続されたセンサー情報を分析するサービスなどがある。データ収集サーバ１１３からのデータの取得は後述のメッセージキューサーバ１２５からメッセージを取得することで行う。

【００２３】

なお、データ利用サーバ１１４は、サーバレスアーキテクチャで実行されるアプリケーションで実現されても良い。具体的には、特定のコンピューティングリソースに対して発生したイベントに応じて、軽量な処理を実行するイベント駆動型コンピューティングサービスという、クラウドコンピューティングサービスで提供されるサービスを指す。例えば、ＡＷＳ Ｌａｍｂｄａ、Ｇｏｏｇｌｅ Ｃｌｏｕｄ Ｆｕｎｃｔｉｏｎｓ、Ｍｉｃｒｏｓｏｆｔ Ａｚｕｒｅ Ｆｕｎｃｔｉｏｎｓ等がある。

【００２４】

第２の認証認可サーバ１２２は、第１の認証認可サーバ１１２とは異なるサーバであり、ストリーム受信サーバ１２３、メッセージサーバ１２４、メッセージキューサーバ１２５への認証・認可を実現し、アクセスを制御するサーバである。

【００２５】

ストリーム受信サーバ１２３は、ストリームによるデータ受信を行うサーバであり、クライアント端末１０２が送信するストリームデータを受信する。

【００２６】

メッセージサーバ１２４は、本システムを構成する各サーバで生成されたイベントをメッセージとして通知するサーバである。メッセージサーバ１２４は、登録された通知先と、通知する条件に基づき、本システムの構成要素のいずれかが生成したイベントを通知先に対してプッシュ通知を行う。

【００２７】

メッセージキューサーバ１２５は、本システムを構成するイベントを送信するサーバからのイベントをメッセージとして格納し、メッセージを受信するサーバに提供するためのサーバである。

【００２８】

特に、ストリーム受信サーバ１２３、データ収集サーバ１１３、およびメッセージサーバ１２４が、本実施例における受信サービスシステムとして機能する。また、メッセージキューサーバ１２５、およびデータ利用サーバ１１４が、本実施例におけるリソースサービスシステムとして機能する。なお、実施例において、クライアント端末１０２及び各サーバ１１２乃至１１４、１２２乃至１２５はそれぞれ１台で図示しているが、複数台で構成されていても良い。

【００２９】

図２は、本実施例に係るクライアント端末１０２、及び各サーバ１１２乃至１１４、１２２乃至１２５を構成する情報処理装置の一般的なハードウェア構成である。なお、各サーバ１１２乃至１１４、１２２乃至１２５は、図２に示すコンピューターの各ハードウェアの機能がそれぞれ、仮想マシンソフトウェアによってアプリケーションソフトウェアとして実現され、物理ハードウェア要素と同様の挙動をとるものとする。

【００３０】

ＣＰＵ２０１は、ＲＯＭ２０３のプログラムＲＯＭに記憶された、あるいはハードディスクなどの外部メモリ２１１からＲＡＭ２０２にロードされたオペレーティングシステムやアプリケーションなどのプログラムを実行する。またＣＰＵ２０１は、システムバス２０４に接続される各ブロックを制御する。後述する各シーケンスの処理は、ＣＰＵ２０１が実行するプログラムによって実現される。

【００３１】

ＲＡＭ２０２は、ＣＰＵ２０１の主メモリ、ワークエリアなどとして機能する。

【００３２】

操作部Ｉ／Ｆ２０５は、操作部２０９からの入力を制御する。

【００３３】
ＣＲＴコントローラ（ＣＲＴＣ）２０６は、ＣＲＴディスプレイ２１０の表示を制御する。
【００３４】
ディスクコントローラ（ＤＫＣ）２０７は各種データを記憶するハードディスクなどの外部メモリ２１１におけるデータアクセスを制御する。
【００３５】
ネットワークコントローラ２０８は、ＷＡＮ１００あるいはＬＡＮ１０１、１０２、１０３を介して接続されたサーバや他の機器との通信制御処理を実行する。
【００３６】
なお、後述する全ての説明においては、特に断りのない限り実行のハードウェア上の主体はＣＰＵ２０１であり、ソフトウェア上の主体は外部メモリ２１１にインストールされたアプリケーションプログラムである。
【００３７】
図３は、本実施例に係るクライアント端末１０２及び各サーバ１１２乃至１１４、１２２乃至１２５のソフトウェア構成を示す図である。各々のモジュールが実行されることで各々の機能を実現する。
【００３８】
図３（Ａ）はクライアント端末１０２のモジュール構成を表す図である。
【００３９】
クライアント端末１０２は、トークンプロバイダモジュール３０１、データ送信モジュール３０２、機器内モジュール３０３、データ保持モジュール３０４、コレクションフィルタ受信モジュール３０５を持つ。
【００４０】
トークンプロバイダモジュール３０１は、第１の認証認可サーバ１１２に対して、クライアント端末の認証要求、アクセストークンの発行依頼や取得を行う。
【００４１】
データ送信モジュール３０２は、データ収集サーバ１１３が提供するサービスを利用したり、クライアント端末１０２のイベントデータをストリーム受信サーバ１２３に送信したりする。データ送信モジュール３０２がストリーム受信サーバ１２３に送信するイベントデータは、後述のデータ保持モジュール３０４に格納されているイベントデータをコレクションフィルタの記述に従って送信する。イベントデータの保持するデータについては後述の図７にて説明を行う。
【００４２】
コレクションフィルタとは、データ保持モジュール３０４で保持されているイベントデータのうち、どのイベントデータをどの頻度で送付するかという送信ルールが定義されているファイルである。コレクションフィルタの保持するデータについては後述の図６にて説明を行う。
【００４３】
機器内モジュール３０３は、クライアント端末１０２を構成するモジュールであり、機器の状態をはじめとした各種イベントデータが発生した際に後述するデータ保持モジュール３０４にイベントデータを格納する。各種イベントデータとは例えばジョブの実行がされたときの開始イベント、終了イベント、またセンサーなどの異常を検知したときの異常イベントなどが挙げられる。
【００４４】
データ保持モジュール３０４は、機器内モジュール３０３からのイベントデータを保持するモジュールである。コレクションフィルタ受信モジュール３０５は、後述するコレクションフィルタ送信モジュール３２３からコレクションフィルタを受信するモジュールである。
【００４５】

第１の認証認可サーバ１１２は、第１の認証認可モジュール３１１、第１のクライアント管理モジュール３１２を持つ。

【００４６】

第１の認証認可モジュール３１１は、クライアント端末１０２からの認証要求に対する処理、認証されたクライアント端末１０２の認可処理を行う。

【００４７】

第１のクライアント管理モジュール３１２は、認証処理及び認可処理を行ったクライアント端末の認証情報としてＩＤやシークレットを管理する。また他のサーバからのリクエストに応じて、管理しているクライアント端末１０２の認証情報を削除するＡＰＩを提供する。

【００４８】

データ収集サーバ１１３は、トークン取得モジュール３２１とストリーム処理モジュール３２２を持つ。

【００４９】

トークン取得モジュール３２１は、データ収集サーバ１１３、クライアント端末１０２がサーバ１２３乃至１２５のサービスを利用するためのトークンを第２の認証認可サーバ１２２から取得するモジュールである。詳しくは後述の図４のシーケンスの中で説明を行う。ストリーム処理モジュール３２２は、ストリーム受信サーバ１２３のストリームデータに対する処理を行うモジュールである。ストリーム処理モジュール３２２は受信したストリームデータの内容を、メッセージサーバ１２４、メッセージキューサーバ１２５を利用してデータ利用サーバ１１４にイベントとして通知する。詳しくは図８のシーケンス、図９のフローチャートを用いて説明を行う。コレクションフィルタ送信モジュール３２３は、クライアント端末１０２へコレクションフィルタを送信するためのモジュールである。

【００５０】

データ利用サーバ１１４は、トークン取得モジュール３３１とメッセージ処理モジュール３３２を持つ。

【００５１】

トークン取得モジュール３３１は、データ利用サーバ１１４がサーバ１２３乃至１２５のサービスを利用するためのトークンを第２の認証認可サーバ１２２から取得するモジュールである。メッセージ処理モジュール３３２は、メッセージキューサーバ１２５に保持されているメッセージを処理するモジュールである。メッセージ処理モジュール３３２は、メッセージを処理することで、ストリーム処理モジュール３２２からのイベントの通知を受信し処理することができる。

【００５２】

第２の認証認可サーバ１２２は、第２の認証認可モジュール３４１、第２のクライアント管理モジュール３４２を持つ。

【００５３】

第２の認証認可モジュール３４１は、クライアントであるクライアント端末１０２や各サーバ１１３、１１４からの認証要求に対する処理、認証されたクライアントの認可処理を行う。第２のクライアント管理モジュール３４２は、認証処理及び認可処理を行った各サーバ１１３、１１４の認証情報としてＩＤやシークレットを管理する。

【００５４】

ストリーム受信サーバ１２３はストリーム受信モジュール３５１を持つ。ストリーム受信モジュール３５１は、ストリーム情報を受信し、保持する機能を提供する。

【００５５】

メッセージサーバ１２４は、メッセージモジュール３６１と送信管理モジュール３６２を持つ。メッセージモジュール３６１は、メッセージの送受信の機能を提供する。送信管理モジュール３６２は、メッセージモジュール３６１が受信したメッセージをどの宛先に送信するかの送信先の管理機能を提供する。

【００５６】
メッセージキューサーバ１２５は、メッセージキューモジュール３７１を持つ。メッセージキューモジュール３７１は受信したメッセージをキューに保持する機能を提供する。
【００５７】
第１の認証認可サーバ１１２は、表Ａのクライアント管理テーブルに示すように、クライアントを一意に識別するためのクライアントＩＤとクライアントを認証するための公開鍵を管理する。さらに第１の認証認可サーバ１１２は、クライアント端末１０２を一意に識別するためのデバイスシリアルを管理する。
【００５８】
第１の認証認可モジュール３１１は、クライアントＩＤと公開鍵を元にクライアント端末１０２を認証し、クライアント端末１０２のデバイスシリアルを特定する。
【００５９】
【表１】

表Ａ：クライアント管理テーブル（第１の認証認可サーバ１１２）

| クライアントＩＤ | 公開鍵 | デバイスシリアル |
|---|---|---|
| ｃｌｉｅｎｔ１ | ｐｕｂｌｉｃｋｅｙ１ | １２３４５６７８９ |
| ｃｌｉｅｎｔ２ | ｐｕｂｌｉｃｋｅｙ２ | ９８７６５４３２１ |

【００６０】
クライアント端末１０２のトークンプロバイダモジュール３０１は表Ｂのクライアント管理テーブルに示すように、自らの端末用のクライアントＩＤと秘密鍵を保持している。
【００６１】
【表２】

表Ｂ：クライアント管理テーブル（クライアント端末１０２）

| クライアントＩＤ | 秘密鍵 |
|---|---|
| ｃｌｉｅｎｔ１ | ｐｒｉｖａｔｅｋｅｙ１ |

【００６２】
本実施例では、クライアントＩＤと非対称鍵を用いた認証を前提として説明するが、前述の通り認証方式は特に問わず、クライアントＩＤとシークレットによる認証といった他の方法でも良い。
【００６３】
同様に、第２の認証認可サーバ１２２は、表Ｃのクライアント管理テーブルに示すように、サーバを一意に識別するクライアントＩＤとサーバを認証するためのシークレットを管理しており、第２の認証認可モジュール３４１はその情報をもとにサーバを認証する。
【００６４】
【表３】

表Ｃ：クライアント管理テーブル（第２の認証認可サーバ１２２）

| クライアントＩＤ | シークレット |
|---|---|
| ｃｌｉｅｎｔＡ | ｓｅｃｒｅｔＡ |
| ｃｌｉｅｎｔＢ | ｓｅｃｒｅｔＢ |

【００６５】
データ収集サーバ１１３のトークン取得モジュール３２１、データ利用サーバ１１４のトークン取得モジュール３３１は、表Ｄのクライアント管理テーブルに示すように、認証を制御すべきクライアントのクライアントＩＤとシークレットを保持している。
【００６６】

【表４】

表Ｄ：クライアント管理テーブル（データ収集サーバ１１３、
データ利用サーバ１１４）

| クライアントＩＤ | シークレット |
|---|---|
| ｃｌｉｅｎｔＡ | ｓｅｃｒｅｔＡ |

【００６７】
図４は、クライアント端末１０２がストリーム受信サーバ１２３へイベントを送信する通常の処理の流れを示したシーケンスである。
【００６８】
Ｓ４０１において、クライアント端末１０２のデータ送信モジュール３０２は、トークンプロバイダモジュール３０１にトークンの発行要求を送信する。トークンプロバイダモジュール３０１は、表Ｂで管理するクライアントＩＤと秘密鍵を利用してアサーションを作成し、アサーションとともにトークン発行リクエストを生成する。本実施例では、アサーションはＲＦＣ７５１９にて決められたＪＳＯＮ　Ｗｅｂ　Ｔｏｋｅｎ（以降、ＪＷＴと称する）を想定しており、ＪＷＴにはクライアントＩＤなどの情報が含まれる。
【００６９】
Ｓ４０２において、トークンプロバイダモジュール３０１は第１の認証認可サーバ１１２の第１の認証認可モジュール３１１にアサーションを送信する。
【００７０】
第１の認証認可モジュール３１１は、Ｓ４０２においてアサーションを送信したクライアント端末に該当する公開鍵を表Ａから取得して、アサーションの署名検証を行う。検証に成功した場合、第１の認証認可モジュール３１１は、アクセストークンＡ１を発行し、レスポンスする。レスポンスを受信したトークンプロバイダモジュール３０１は、アクセストークンＡ１をデータ送信モジュール３０２にレスポンスする。アクセストークンＡ１は、第１の認証認可サーバ１１２によってアクセス制御が行われるデータ収集サーバ１１３へのクライアント端末１０２がアクセスするためのものである。
【００７１】
本実施例では、アクセストークンはＪＷＴを想定しており、クライアントＩＤやクライアント端末１０２のデバイスシリアルといった情報、トークンの有効期限などの情報が含まれる。
【００７２】
Ｓ４０３において、データ送信モジュール３０２はＳ４０２のレスポンスとして受信したアクセストークンＡ１を利用して、データ収集サーバ１１３のトークン取得モジュール３２１にトークン発行要求を送信する。
【００７３】
Ｓ４０４において、トークン取得モジュール３２１は受信したトークン発行要求に含まれるアクセストークンＡ１の検証を行う。トークン取得モジュール３２１は第１の認証認可モジュール３１１が発行したＪＷＴを検証するための公開鍵を保持しており、ＪＷＴを検証することでクライアントの認可を行う。ＪＷＴの検証では、トークン取得モジュール３２１はＪＷＴの署名が正しいかを公開鍵を用いて検証し、さらにＪＷＴの有効期間中かどうかを検証する。またアクセストークンＡ１を検証した結果、クライアント情報を取得することが可能であり、アクセス元のクライアント端末１０２をデバイスシリアルで識別できる。なお、トークン取得モジュール３２１は第１の認証認可モジュール３１１にトークンの検証を依頼し、検証結果とクライアント情報を取得するといった別の実施形でも良い。リクエストを受けたトークン取得モジュール３２１はＳ４０５、Ｓ４０６において受信したリクエストに対する処理を行う。
【００７４】
Ｓ４０５において、トークン取得モジュール３２１は第２の認証認可サーバ１２２の第

２の認証認可モジュール３４１にトークン発行要求を送信する。トークン発行要求は表Ｄのクライアント ＩＤとシークレットとともに送信する。トークン発行要求を受信した第２の認証認可モジュール３４１は、表Ｃで管理するクライアントＩＤとシークレットと、トークン発行要求に含まれるクライアントＩＤとシークレットとが一致する場合には、アクセストークンＢを発行してレスポンスする。アクセストークンＢは、第２の認証認可サーバ１２２にアクセスするためのものである。

【００７５】

Ｓ４０６において、アクセストークンＢを取得したトークン取得モジュール３２１は、第２の認証認可モジュール３４１に対して、ＩＤトークン発行要求を送信する。ＩＤトークン発行要求では、トークン取得モジュール３２１はアクセストークンＢとＳ４０４の検証結果から取得したクライアント端末１０２のクライアントＩＤを送信する。第２の認証認可モジュール３４１は、受信したクライアントＩＤ用のＩＤトークンを発行する。

【００７６】

ＩＤトークンとは、クライアント端末１０２を確かに認証したことを証明するトークンである。本実施例では、クライアント端末１０２の認証は、Ｓ４０４においてトークン取得モジュール３２１がアクセストークンＡ１を検証することで確認している。第２の認証認可モジュール３４１はトークン取得モジュール３２１に対してクライアントＩＤとシークレットを提供する形で信頼関係を結んでおり、トークン取得モジュール３２１がクライアント端末１０２を認証した結果を信頼する。第２の認証認可モジュール３４１はその信頼をもとにＩＤトークンを発行している形となる。また、ＩＤトークンはＪＷＴ形式を想定しており、第２の認証認可モジュール３４１は署名を検証することでＩＤトークンを検証可能である。

【００７７】

Ｓ４０７において、ＩＤトークンを受信したクライアント端末１０２のデータ送信モジュール３０２は、第２の認証認可サーバ１２２の第２の認証認可モジュール３４１にトークン発行要求を行う。この要求にはＳ４０４で取得したＩＤトークンを含める。要求を受けた第２の認証認可モジュール３４１はＩＤトークンの検証を行い、アクセストークンＣを発行しレスポンスする。アクセストークンＣは、ストリーム受信サーバ１２３へアクセスするためのものである。

【００７８】

Ｓ４０８において、アクセストークンＣを受信したクライアント端末１０２のデータ送信モジュール３０２は、ストリーム受信サーバ１２３のストリーム受信モジュール３５１にデータ送信を行う。本データ送信ではアクセストークンＣと、アクセストークンＡ１を送信する。データ送信要求を受信したストリーム受信モジュール３５１はアクセストークンＣを検証して問題が無ければ、アクセストークンＡ１とデータのペアを保持する。

【００７９】

以上により、クライアント端末１０２はストリーム受信サーバ１２３へのデータ送信を完了する。

【００８０】

図４で示したＳ４０８にて説明を行ったデータ送信処理について、図５のフローチャートを用いて詳細に説明を行う。

【００８１】

図５はコレクションフィルタ６０１に従ってストリーム受信モジュール３５１にイベントデータを送信するデータ送信モジュール３０２の処理のフローチャートである。

【００８２】

コレクションフィルタ６０１は図６に示すようなデータである。図６（Ａ）は通常モード時のコレクションフィルタ、図６（Ｂ）はセーフモード時のコレクションフィルタの例である。コレクションフィルタ６０１には、イベントが発生したエンティティ６０２と発生したイベント名６０３に対してストリーム受信モジュール３５１へ送信する送信タイミング６０４、コレクションフィルタモード６０５が記載されている。コレクションフィル

タモード６０５には通常モードかセーフモードであるかが記載される。
【００８３】
　また図６（Ｂ）にようにコレクションフィルタモード６０５がセーフモードの場合には、さらに閾値６０６が記載される。
【００８４】
　図６（Ａ）の例では、ＨＤＤのセンサーエラーが発生した際には１０分おきにイベントデータを送信するというフィルタ設定であり、コレクションフィルタモード６０５はノーマルモードであることを示す。
【００８５】
　図６（Ｂ）の例では、ＨＤＤのセンサーエラーのコレクションフィルタモード６０５をセーフモードにしており、６０分おきにチェックをし、イベントデータの件数が閾値の５を超えていない場合にイベントデータ送信するというフィルタ設定であることを示す。
【００８６】
　図５のフローチャートは、データ送信モジュール３０２がコレクションフィルタ６０１の記述に従って、イベントデータを送信するタイミングで実行される。実行時には引数として実行するコレクションフィルタ６０１のエンティティ６０２とイベント名６０３とコレクションフィルタモード６０５が渡される。
【００８７】
　Ｓ５０１において、データ送信モジュール３０２は引数で渡されたエンティティ、イベント名、コレクションフィルタモードを取得する。
【００８８】
　Ｓ５０２において、データ送信モジュール３０２はデータ保持モジュール３０４で保持されているイベントデータ７０１から、Ｓ５０１で取得したエンティティとイベント名と一致するイベントデータ一覧を取得する。イベントデータ７０１は図７（Ａ）に示すようなデータであり、クライアント端末１０２内でのイベントを一意に識別するイベントＩＤ７０２、クライアント端末１０２を一意に識別するデバイスシリアル７０３が記載されている。さらにイベントデータ７０１には、イベントが発生したエンティティ７０４、発生したイベント名７０５と発生日時７０６が記載されている。Ｓ５０２では、Ｓ５０１で取得したエンティティとイベント名と一致するエンティティ７０４、イベント名７０５のイベントデータ一覧を、送信対象のデータとして取得する。
【００８９】
　Ｓ５０３において、データ送信モジュール３０２はＳ５０１で取得したコレクションフィルタモード６０５の値がセーフモードかどうかをチェックする。セーフモードであった場合にはＳ５０６に進み、異なる場合にはＳ５０４に進む。
【００９０】
　Ｓ５０４において、データ送信モジュール３０２はＳ５０２で取得したイベントデータ一覧をストリーム受信モジュール３５１にアクセストークンＣと、アクセストークンＡ１とともに送信する。
【００９１】
　Ｓ５０５において、データ送信モジュール３０２はＳ５０４で送信したイベントデータ一覧をデータ保持モジュール３０４から削除する。
【００９２】
　Ｓ５０６において、データ送信モジュール３０２はＳ５０２で取得したイベントデータ一覧の件数がコレクションフィルタ６０１の閾値６０６で示される所定値を超えて発生しているかをチェックする。ここでの所定値は、ストリーム受信モジュール３５１が単位時間あたりに受信可能なデータ量に基づき定められる。所定値を超えている場合にはＳ５０９に進み、所定値以下である場合にはＳ５０７に進む。なお、イベントデータの数ではなく、イベントデータのサイズを、Ｓ５０６での判断に用いるデータ量として扱っても良い。
【００９３】

Ｓ５０７において、データ送信モジュール３０２は閾値を超えていないためクライアント端末１０２は安定したと判定する。そして、データ送信モジュール３０２は、セーフモードのコレクションフィルタから、クライアント端末１０２の外部メモリ２１１に退避していた通常モードのコレクションフィルタに変更する。

【００９４】

Ｓ５０８において、データ送信モジュール３０２はセーフモードのコレクションフィルタから通常モードのコレクションフィルタに変更したことを通知するためのセーフコレクションフィルタ設定解除イベントを新規に作成する。そして、データ送信モジュール３０２は、セーフコレクションフィルタ設定解除イベントを、アクセストークンＣおよびアクセストークンＡ１とともにストリーム受信モジュール３５１に送信する。

【００９５】

本実施例ではコレクションフィルタ設定解除イベントは、図７（Ｂ）のようにセーフモード７０７の値に「ｒｅｌｅａｓｅ」と記載された項目を付与したものとする。

【００９６】

なお、本実施例では、Ｓ５０７にようにクライアント端末が通常のコレクションフィルタに変更する方法を説明した。しかしながら、単位時間当たりのイベント発生数が閾値以下になったことをクライアント端末がストリーム受信サーバ１２３に対して通知し、データ収集サーバ１１３がクライアント端末に対して通常のコレクションフィルタへ変更するように要求してもよい。

【００９７】

Ｓ５０９において、データ送信モジュール３０２は閾値を超えているためクライアント端末１０２は安定していないと判定し、Ｓ５０２で取得したイベントデータ一覧は送信せずに削除する。ここでは、クライアント端末における記憶容量には限りがあるため、送信しないイベントデータは削除されるものとしているが、送信しないイベントデータは必ずしも削除されなくても良い。なお、データ送信モジュール３０２は、イベントデータ一覧のデータ全てを送信せずに削除するのではなく、一部のデータを送信せずに削除して残りのデータをストリーム受信モジュール３５１に送信しても良い。ただし、その場合は、ストリーム受信モジュール３５１に送信されるイベントのデータ数は閾値以下となるように、一部のデータを削除するものとする。それにより、ストリーム受信モジュールの負荷を下げるという効果を得られる。

【００９８】

以上、説明した図５のフローチャートによって、コレクションフィルタがセーフモードに設定されている際には、イベントデータが閾値を超えている場合にはイベントデータを送信しないことで、ストリーム受信モジュールの負荷を下げることができる。またイベントデータが閾値を超えなかった場合には通常モードのコレクションフィルタに戻すことにより、クライアント端末１０２が安定した際にデータの送付を自動で再開することができる。

【００９９】

図４、図５で示した処理が完了した際にストリーム受信サーバ１２３が保持するデータの一例を表Ｅで示す。ストリーム受信サーバ１２３においてデータを一意に識別するＩＤ、データを受信した受信日時、受信したアクセストークン、デバイスシリアル、エンティティ、イベント名、セーフモードの解除イベントかどうかの情報が１レコードとして管理される。ストリームデータ保持テーブルでは、受信日時順にソートされてデータが保存されている。

【０１００】

【表５】

表Ｅ：ストリームデータ保持テーブル

| ＩＤ | 受信日時 | アクセストークン | デバイスシリアル | エンティティ | イベント名 | セーフモード |
|---|---|---|---|---|---|---|
| 1 | ２０１８／０８／３１００：００：０１ | Ａ１ | ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ | |
| 2 | ２０１８／０８／３１００：００：０２ | Ａ１ | ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ | |
| 3 | ２０１８／０８／３１００：００：０３ | Ａ１ | ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ | |
| 4 | ２０１８／０８／３１００：００：０５ | Ａ１ | ＡＡＡ１２３４５ | Ｐｒｉｎｔ | ｊｏｂＳｔａｒｔ | |
| 5 | ２０１８／０８／３１００：００：０６ | Ａ１ | ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ | |
| 6 | ２０１８／０９／０５１３：０４：００ | Ａ１ | ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ | 解除 |

【０１０１】

以降図８、図９を用いてデータ収集サーバ１１３のストリーム処理モジュール３２２での処理についての説明を行う。

【０１０２】

図８と図９はそれぞれデータ収集サーバ１１３のストリーム処理モジュール３２２が提供する処理の一連の流れを示したシーケンスとフローチャートである。本処理の詳細な説明は図９のフローチャートを用いて説明し、その他のモジュールとの連携部分についてはシーケンスを用いて説明する。本処理についてはデータ収集サーバ１１３によって定期的に実行される処理である。

【０１０３】

Ｓ９０１（Ｓ８０１）において、ストリーム処理モジュール３２２は、第２の認証認可モジュール３４１にトークン発行要求を送信する。本処理はＳ４０５と同様の処理である。第２の認証認可モジュール３４１は、トークン発行要求を受信して、ストリーム受信サーバ１２３、メッセージサーバ１２４にアクセスするために必要なアクセストークンＢを発行しストリーム処理モジュール３２２にレスポンスする。

【０１０４】

Ｓ９０２（Ｓ８０２）において、ストリーム処理モジュール３２２はストリーム受信サーバ１２３のストリーム受信モジュール３５１にレコード取得をアクセストークンＢとともに要求する。レコード取得要求を受信したストリーム受信モジュール３５１は、アクセストークンＢを検証し、問題がなければ表Ｅのストリームデータ保持テーブルで管理されるレコードから受信日の古いレコードから順にストリーム処理モジュール３２２にレスポンスする。受信したレコードを移行イベントレコードと称す。

【０１０５】

Ｓ９０３において、ストリーム処理モジュール３２２はＳ９０２で受信したイベントレコードからアクセストークンＡ１を取得する。Ｓ９０４において、ストリーム処理モジュール３２２はＳ９０３で取得したアクセストークンＡ１の検証を行う。検証ではアクセストークンＡ１の署名、有効期間のチェックを行う。Ｓ９０５において、ストリーム処理モジュール３２２はＳ９０４の検証で問題がなければＳ９０６に進む。問題がある場合には処理を終了する。

【０１０６】

Ｓ９０６において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントレコードのデバイスシリアルを取得する。Ｓ９０７において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントレコードのエンティティを取得する。Ｓ９０８において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントレコードのイベント

名を取得する。

【０１０７】

Ｓ９０９において、ストリーム処理モジュール３２２は、Ｓ９０６乃至Ｓ９０８で取得したデバイスシリアル、エンティティ、イベント名と一致するデータがセーフモード設定イベントテーブルに登録されているかどうかをチェックする。セーフモード設定イベントテーブルは、データ収集サーバ１１３に格納されている。もし登録されている場合にはＳ９１７に進み、登録されていない場合にはＳ９１０に進む。

【０１０８】

セーフモード設定イベントテーブルとはクライアント端末１０２でセーフモードのコレクションフィルタの設定依頼を出しているかどうかを管理しているテーブルであり、その一例を表Ｆで示す。

【０１０９】

【表６】

表Ｆ：セーフモード設定イベントテーブル（データ収集サーバ１１３）

| デバイスシリアル | エンティティ | イベント名 |
|---|---|---|
| ＡＡＡ１００００ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ |
| ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ |
| ＢＢＢ２３５４１ | ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ |

【０１１０】

Ｓ９１０において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントデータから受信日時を取得する。

【０１１１】

Ｓ９１１において、ストリーム処理モジュール３２２は、特定のクライアント端末１０２で同じイベントが大量に発生していないかの流量チェックを行う。流量チェックの一例として、図９（Ｂ）のフローチャートを用いて説明する。

【０１１２】

Ｓ９３１において、ストリーム処理モジュール３２２は、Ｓ９０６乃至Ｓ９０８およびＳ９１０で取得したデバイスシリアル、エンティティ、イベント名、受信日時を１つのレコードとしてイベント履歴テーブルに追加する。

【０１１３】

イベント履歴テーブルとは、過去に処理したイベントの履歴を管理するテーブルであり、その１例を表Ｇで示す。このイベント履歴テーブルのレコードは受信日時から１時間経過したものは定期的に削除される。

【０１１４】

【表７】

表Ｇ：イベント履歴テーブル（データ収集サーバ１１３）

| デバイスシリアル | エンティティ | イベント名 | 受信日時 |
|---|---|---|---|
| ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒ Ｅｒｒｏｒ | ２０１８／０８／３１ ００：００：０１ |
| ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒ Ｅｒｒｏｒ | ２０１８／０８／３１ ００：００：０２ |
| ＡＡＡ１２３４５ | ＨＤＤ | ｓｅｎｓｏｒ Ｅｒｒｏｒ | ２０１８／０８／３１ ００：００：０３ |

【０１１５】

Ｓ９３２において、ストリーム処理モジュール３２２はＳ９０６乃至Ｓ９０８で取得したデバイスシリアル、エンティティ、イベント名と一致するイベント履歴一覧を表Ｇのイ

ベント履歴テーブルから取得する。

【０１１６】

Ｓ９３３において、ストリーム処理モジュール３２２はＳ９３２で取得したイベント履歴一覧で単位時間あたりに発生しているイベント数が閾値を超えているかをチェックする。例えば１分間に同じイベントが１００を超えて発生しているなどをチェックする。ここでの閾値は、ストリーム受信モジュール３５１が単位時間あたりに受信可能なデータ量に基づき定められる。もし閾値を超えている場合にはＳ９３４で閾値を超えているという戻り値を返す、超えていない場合にはＳ９３５で閾値を超えていないという戻り値を返す。

【０１１７】

図９（Ａ）の説明に戻る。Ｓ９１２において、ストリーム処理モジュール３２２はＳ９１１の結果として閾値を超えていると判断された場合にはＳ９１４に進み、異なる場合はＳ９１３に進む。

【０１１８】

Ｓ９１３において、ストリーム処理モジュール３２２は特定のクライアント端末１０２で同じイベントが大量に発生していないと判定し、データ利用サーバ１１４に対して通常のイベント通知処理を行う。

【０１１９】

具体的にはＳ８０３においてストリーム処理モジュール３２２はメッセージサーバ１２４に対して発生したイベントとアクセストークンＢを送信する。Ｓ８０４において、メッセージサーバ１２４のメッセージモジュール３６１はＳ８０６で受信したアクセストークンを検証し、問題がなければイベントを送信管理モジュール３６２の設定に従ってメッセージキューサーバ１２５に送信して処理を終了する。

【０１２０】

送信管理モジュール３６２が保持する送信管理データの一例を表Ｈに示す。

【０１２１】

【表８】

表Ｈ：送信管理データテーブル（メッセージサーバ１２４）

| トピックＩＤ | 通知先クライアントＩＤ |
|---|---|
| ＨＤＤ | ｃｌｉｅｎｔ１ |
| Ｐｒｉｎｔ | ｃｌｉｅｎｔ１ |

【０１２２】

送信管理データはメッセージの格納先を一意に識別するトピックＩＤ、そのメッセージの通知先となるクライアントＩＤとを関連付けて１レコードとして保存される。

【０１２３】

またメッセージモジュール３６１が保存するデータの一例を表Ｉに示す。

【０１２４】

【表９】

表Ｉ：メッセージテーブル（メッセージサーバ１２４）

| トピックＩＤ | メッセージ |
|---|---|
| ＨＤＤ | ｓｅｎｓｏｒＥｒｒｏｒ |

【０１２５】

メッセージモジュール３６１は受信したイベントのメッセージを該当するトピックＩＤと関連付けて１つのレコードとしてメッセージテーブルに保存される。

【０１２６】

図８および図９の説明に戻る。Ｓ９１４（Ｓ８０５）において、ストリーム処理モジュール３２２は、特定のクライアント端末１０２で同じイベントが大量に発生していると判

定する。そして、ストリーム処理モジュール３２２は、クライアント端末１０２に対してセーフモードのコレクションフィルタにするようコレクションフィルタ送信モジュール３２３に要求する。

【０１２７】

Ｓ８０６において、コレクションフィルタ送信モジュール３２３はクライアント端末１０２のコレクションフィルタ受信モジュール３０５に対してセーフモードのコレクションフィルタを送信する。セーフモードのコレクションフィルタは図６（Ｂ）のようなコレクションフィルタ６０１である。コレクションフィルタ送信モジュール３２３は、Ｓ８０２で取得したエンティティ、イベント名に対してコレクションフィルタモード６０５をセーフモード、送信タイミング６０４、閾値６０６に値を設定したものを送付する。

【０１２８】

なお、本実施例では、上述したようにコレクションフィルタ送信モジュール３２３がコレクションフィルタをコレクションフィルタ受信モジュール３０５に送信する記載にしている。他の実施の形態としては、コレクションフィルタ送信モジュール３２３は、クライアント端末の状態情報を管理する不図示の状態管理サーバに対してコレクションフィルタを設定するように指示しておく。そして、クライアント端末と状態管理サーバで管理するクライアント端末の状態情報とを同期することで、予め設定されたコレクションフィルタがコレクションフィルタ受信モジュール３０５に対して送信される。これにより、コレクションフィルタの設定変更をクライアント端末に適用させることができる。例えば、Ａｍａｚｏｎ Ｗｅｂ Ｓｅｒｖｉｃｅ（以降、ＡＷＳと称する）が提供するサービスであるＡＷＳ ＩｏＴにはＤｅｖｉｃｅ Ｓｈａｄｏｗというサービスを利用することで、この処理を実現してもよい。

【０１２９】

本実施の形態では送信タイミング６０４は６０ｍｉｎ、閾値を５という固定の値を設定することで説明を行うが、エンティティ、イベント名に応じて値を変更してもよい。

【０１３０】

Ｓ８０７において、コレクションフィルタ受信モジュール３０５はＳ８０６で受信したセーフモードのコレクションフィルタ６０１をデータ送信モジュール３０２に設定する。その際に以前使用していたコレクションフィルタは外部メモリ２１１に退避しておき、クライアント端末１０２の挙動が安定した際には再度使用できるようにしておく。

【０１３１】

Ｓ９１５において、ストリーム処理モジュール３２２は特定のクライアント端末１０２で同じイベントが大量に発生していることをデータ利用サーバ１１４に対して伝えるためにセーフモードのコレクションフィルタイベント送信処理を行う。本処理を行うことで、例えば、データ利用サーバ１１４がクライアント端末に対して提供しているＵＩ画面に対象のデバイスが異常状態である旨を表示することができる。

【０１３２】

具体的にはＳ８０８において、ストリーム処理モジュール３２２は、セーフモードのコレクションフィルタに設定したというセーフコレクションフィルタ設定イベントをメッセージサーバ１２４のメッセージモジュール３６１に送信する。これは、特定のクライアント端末１０２で同じイベントが大量に発生していることをデータ利用サーバ１１４に対して伝えるためである。セーフコレクションフィルタ設定イベントの一例で、通常のイベントのメッセージ「ｓｅｎｓｏｒＥｒｒｏｒ」の場合、セーフコレクションフィルタ設定イベントは「ＳＡＦＥＭＯＤＥ＿ｓｅｎｓｏｒＥｒｒｏｒ」というメッセージにする。これにより、ストリーム処理モジュール３２２は、ＨＤＤのｓｅｎｓｏｒＥｒｒｏｒイベントがセーフモードになったことを通知することができる。

【０１３３】

Ｓ８０９において、メッセージサーバ１２４のメッセージモジュール３６１はＳ８１１で受信したイベントを送信管理モジュール３６２の設定に従ってメッセージキューサーバ１２５に送信する。

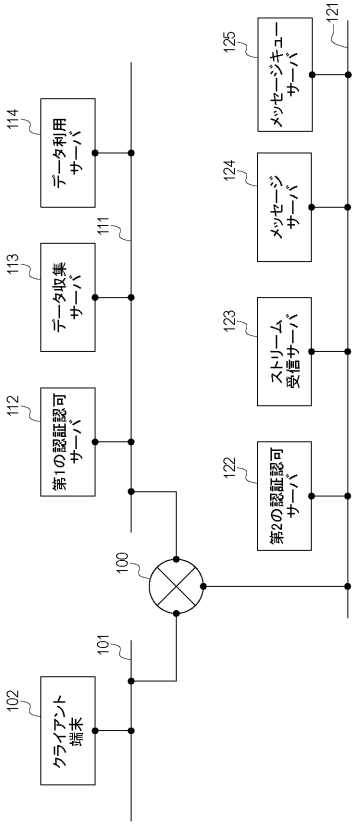
【０１３４】
Ｓ９１６において、ストリーム処理モジュール３２２はＳ８０２で取得したイベントレコードからデバイスシリアル、エンティティ、イベント名を表Ｆのセーフモード設定イベントテーブルのレコードとして登録して処理を終了する。
【０１３５】
Ｓ９１７は、Ｓ９０９でデータがセーフモード設定ベントテーブルに登録されている場合に行われる処理である。Ｓ９１７において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントレコードのセーフモードの値を取得する。
【０１３６】
Ｓ９１８において、ストリーム処理モジュール３２２はＳ９１７で取得したセーフモード値が「解除」であるかをチェックする。もし解除である場合にはＳ９１９に進み、解除でない場合には処理を終了する。
【０１３７】
Ｓ９１９において、ストリーム処理モジュール３２２は、セーフモードを解除したことをデータ利用サーバ１１４に対して伝えるためにセーフモードのコレクションフィルタの設定を解除したというセーフコレクションフィルタ解除設定イベント送信処理を行う。
【０１３８】
具体的にはＳ８１０において、ストリーム処理モジュール３２２はセーフコレクションフィルタ解除設定イベントをメッセージサーバ１２４のメッセージモジュール３６１に送信する。
【０１３９】
セーフコレクションフィルタ解除設定イベントの一例で、通常のイベントのメッセージ「ｓｅｎｓｏｒＥｒｒｏｒ」の場合、セーフコレクションフィルタ設定イベントは「ＢＡＳＩＣＭＯＤＥ＿ｓｅｎｓｏｒＥｒｒｏｒ」というメッセージにする。これにより、ＨＤＤのｓｅｎｓｏｒＥｒｒｏｒイベントが通常モードになったことを通知できる。
【０１４０】
Ｓ８１１において、メッセージサーバ１２４のメッセージモジュール３６１はＳ８１０で受信したイベントを送信管理モジュール３６２の設定に従ってメッセージキューサーバ１２５に送信する。
【０１４１】
Ｓ９２０において、ストリーム処理モジュール３２２はＳ９０９で取得したセーフモード設定イベントレコードを削除して処理を終了する。
【０１４２】
以上説明した図８のシーケンス、図９のフローチャートにより、クライアント端末１０２からの想定外の大量データが送付されてきた場合にはクライアント端末１０２に対してセーフモードのコレクションフィルタを設定させる。クライアント端末１０２は、コレクションフィルタが設定されたことにより、受信サービスシステムに送信するデータの量を削減する。またクライアント端末１０２のデータ送付が閾値以下になったことを示すコレクションフィルタ設定解除イベントを受信サービスが受信した際には、セーフモードが解除されたことをサブスクライバであるデータ利用サーバ１１４に通知する。
【０１４３】
このことにより、一部のクライアント端末から過剰なデータが送付されてきた場合であっても、データ収集サーバ１１３の機能を停止させずに継続してサービスを提供し続けることができる。つまり、データ収集サーバ１１３における他のクライアント端末のデータの処理が遅延することを回避できる。またサブスクライバに対しても現在どのような状況であるのかを通知することで、その状態をユーザに通知するなどを行うことができる。
【０１４４】
（実施例２）
実施例１で説明した第１の実施の形態により、セーフモードのコレクションフィルタ６０１を設定されたクライアント端末１０２は、コレクションフィルタの設定によってイベ

ントデータの送信頻度が閾値以下になるまで送信しないようにできる。
【０１４５】
しかし、クライアント端末１０２の故障の状態によってはセーフモードのコレクションフィルタを受け付けないことも考えられる。本実施例で説明する第２の実施の形態は上記のような状態を考慮したものである。
【０１４６】
図１０は第２の実施の形態におけるデータ収集サーバ１１３のストリーム処理モジュール３２２での処理のフローチャートである。
【０１４７】
Ｓ９０１乃至Ｓ９１７、Ｓ９１９とＳ９２０の処理は図９で説明したものと同じである。
【０１４８】
Ｓ１００１において、ストリーム処理モジュール３２２はＳ９１７で取得したセーフモード値が「解除」であるかをチェックする。もし解除である場合にはＳ９１９に進み、解除でない場合にはＳ１００２に進む。
【０１４９】
Ｓ１００２において、ストリーム処理モジュール３２２はＳ９１４で依頼したセーフモードコレクションフィルタ設定要求処理から一定時間経過しているかを確認する。一定期間経過している場合にはＳ１００３に進み、経過していない場合には処理を終了する。一定期間は本実施の形態では「１０分」という固定の値とする。
【０１５０】
Ｓ１００３において、ストリーム処理モジュール３２２はＳ９０６で取得したデバイスシリアルのクライアント端末１０２の認証情報を削除するように第１の認証認可サーバ１１２の第１のクライアント管理モジュール３１２に対して依頼をする。
【０１５１】
以上、本実施例で説明した第２の実施の形態により、セーフモードコレクションフィルタ設定要求を出したクライアント端末１０２から一定の期間がたってもデータ送信頻度が下がらない場合には、そのクライアント端末の認証情報が削除されるようにした。そうすることでクライアント端末１０２がＳ４０２にてアクセストークンの取得を行おうとした際に認証できなり、データ収集サーバ１１３へのアクセスする処理まで到達できなくなる。すなわち、クライアント端末１０２は受信サービスに対してデータを送信できなくなるため、データ収集サーバ１１３の負荷を下げることができる。
【０１５２】
（実施例３）
第１の実施の形態、第２の実施の形態はデータ収集サーバ１１３がシステムとして許容できるレベルのアクセスに対してはできるだけ受信しようとするものであった。
【０１５３】
しかしながら、顧客のネットワークの負荷を考慮した場合、故障だと思われるイベントデータの送信を検知した場合に実施例１，２のデータ送信数になるよりも少ない量でもデバイス側でセーフモードのコレクションフィルタを設定したいことも考えられる。
【０１５４】
第３の実施の形態では上記のケースを考慮して、クライアント端末側でコレクションフィルタを切り替えることを考慮したものである。
【０１５５】
図１１は第３の実施の形態におけるコレクションフィルタ６０１に従ってストリーム受信モジュール３５１にイベントデータを送信するデータ送信モジュール３０２の処理のフローチャートである。
【０１５６】
Ｓ５０１、Ｓ５０２、Ｓ５０４乃至Ｓ５０９の処理は図５で説明したものと同じである。

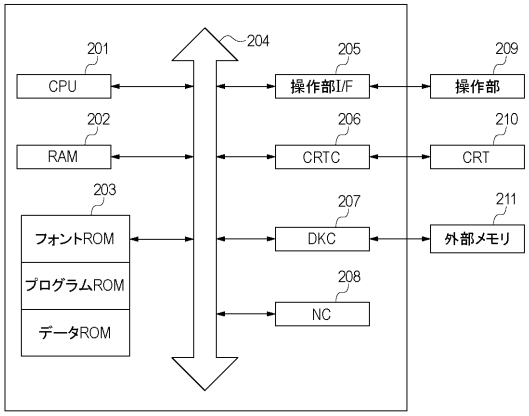
【０１５７】
Ｓ１１０１において、データ送信モジュール３０２はＳ５０１で取得したコレクションフィルタモード６０５の値がセーフモードかどうかをチェックする。セーフモードであった場合にはＳ５０６に進み、異なる場合にはＳ１１０２に進む。
【０１５８】
Ｓ１１０２において、データ送信モジュール３０２はＳ５０２で取得したイベントデータ一覧がセーフモードに移行する閾値を超えているかをチェックする。セーフモードに移行する閾値は本実施の形態では固定で１００００とするが、イベントデータ毎に閾値を設けるなどをしてもよい。もし、閾値を超えている場合にはＳ１１０３に進み、超えていない場合にはＳ５０４に進む。
【０１５９】
Ｓ１１０３において、データ送信モジュール３０２はセーフモードに移行するためセーフモードコレクションフィルタに設定する。本実施の形態に置いてこのセーフモードコレクションフィルタと通常のコレクションフィルタは事前にクライアント端末１０２に保持されているものとする。
【０１６０】
Ｓ１１０４において、データ送信モジュール３０２は通常モードのコレクションフィルタからセーフモードのコレクションフィルタに変更したことを通知するためのセーフコレクションフィルタ設定イベントを新規に作成する。データ送信モジュール３０２は、作成したセーフコレクションフィルタ設定イベントを、アクセストークンＣおよびアクセストークンＡ１とともにストリーム受信モジュール３５１に送信する。
【０１６１】
本実施例ではコレクションフィルタ設定イベントは、図７（Ｂ）のセーフモード７０７の値に「ｓｅｔ」と記載された項目を付与したものとする。
【０１６２】
またコレクションフィルタ設定イベントを受信したストリームデータ保持テーブルのセーフモードの値には「設定」が入るものとする。
【０１６３】
以上説明した図１１のフローチャートによりクライアント端末１０２側での異常を検知した際に自動でセーフモードのコレクションフィルタに変更し、変更したことをデータ収集サーバ１１３に通知することができる。
【０１６４】
図１２は第３の実施の形態におけるデータ収集サーバ１１３のストリーム処理モジュール３２２での処理のフローチャートである。
【０１６５】
Ｓ９０１乃至Ｓ９０８、Ｓ９１０乃至Ｓ９１７、Ｓ９１９とＳ９２０の処理は図９で説明したものと同じである。またＳ１００１乃至Ｓ１００３の処理は図１０で説明したものと同じである。
【０１６６】
Ｓ１００１において、ストリーム処理モジュール３２２は、Ｓ９０６乃至Ｓ９０８で取得したデバイスシリアル、エンティティ、イベント名と一致するデータがセーフモード設定イベントテーブルに登録されているかどうかをチェックする。セーフモード設定イベントテーブルは、データ収集サーバ１１３に格納されている。もし登録されている場合にはＳ９１７に進み、登録されていない場合にはＳ１２０２に進む。
【０１６７】
Ｓ１００２において、ストリーム処理モジュール３２２はＳ９０２で取得したイベントレコードのセーフモードの値を取得する。
【０１６８】
Ｓ１００３において、ストリーム処理モジュール３２２はＳ１００２で取得したセーフモード値が「設定」であるかをチェックする。

【０１６９】
「設定」である場合にはＳ９１５に進み、異なる場合にはＳ９１０に進む。
【０１７０】
以上説明した図１２のフローチャートにより、データ収集サーバ１１３はクライアント端末側でセーフモードのコレクションフィルタに設定したという通知を受信した際に、それをデータ利用サーバ１１４に通知することができる。
【０１７１】
以上、説明した第３の実施の形態によりクライアント端末１０２側で故障と判断した場合にセーフモードのコレクションフィルタを適用することによってイベントの送信数を抑えることができる。これにより顧客環境でのネットワーク負荷を考慮したデータコレクションによる送信機能を提供することができる。またセーフモードに移行したことをデータ収集サーバ１１３へ通知することで、データ利用サーバ１１４へもセーフモードのコレクションフィルタに移行したことを通知することができる。
【０１７２】
（他の実施例）
本発明は、上述した実施形態を適宜組み合わせることにより構成された装置あるいはシステムやその方法も含まれるものとする。
【０１７３】
ここで、本発明は、上述した実施形態の機能を実現する１つ以上のソフトウェア（プログラム）を実行する主体となる装置あるいはシステムである。また、その装置あるいはシステムで実行される上述した実施形態を実現するための方法も本発明の１つである。また、そのプログラムは、ネットワークまたは各種記憶媒体を介してシステムあるいは装置に供給され、そのシステムあるいは装置の１つ以上のコンピューター（ＣＰＵやＭＰＵ等）によりそのプログラムが読み出され、実行される。つまり、本発明の１つとして、さらにそのプログラム自体、あるいは当該プログラムを格納したコンピューターにより読み取り可能な各種記憶媒体も含むものとする。また、上述した実施形態の機能を実現する回路（例えば、ＡＳＩＣ）によっても、本発明は実現可能である。
【符号の説明】
【０１７４】
１０２　クライアント端末
１１３　データ収集サーバ
１１４　データ利用サーバ
１２３　ストリーム受信サーバ
１２４　メッセージサーバ
１２５　メッセージキューサーバ

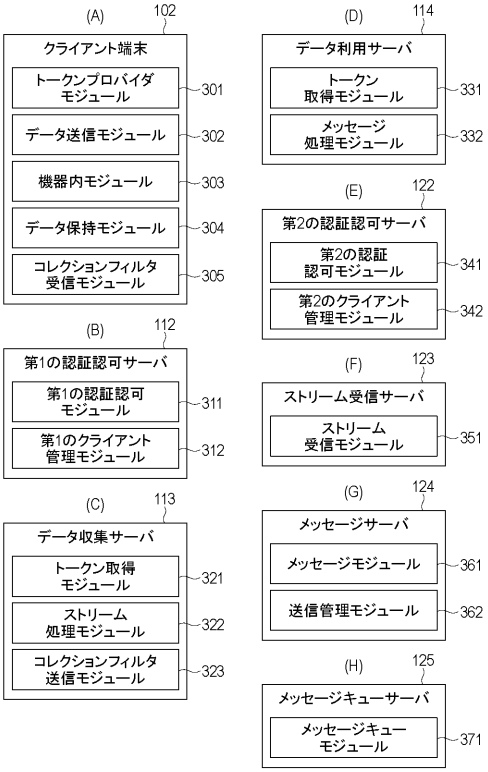
【図１】


データ利用サーバ
114
データ収集サーバ
113
111
第1の認証認可サーバ
112
メッセージキューサーバ
125
121
メッセージサーバ
124
ストリーム受信サーバ
123
第2の認証認可サーバ
122
100
101
クライアント端末
102

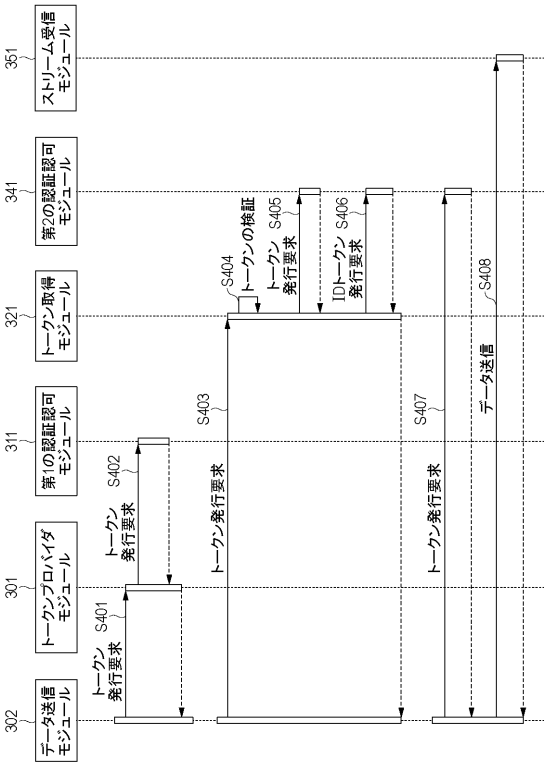
【図２】


CPU
201
RAM
202
フォントROM
203
プログラムROM
データROM
204
操作部I/F
205
CRTC
206
DKC
207
NC
208
操作部
209
CRT
210
外部メモリ
211

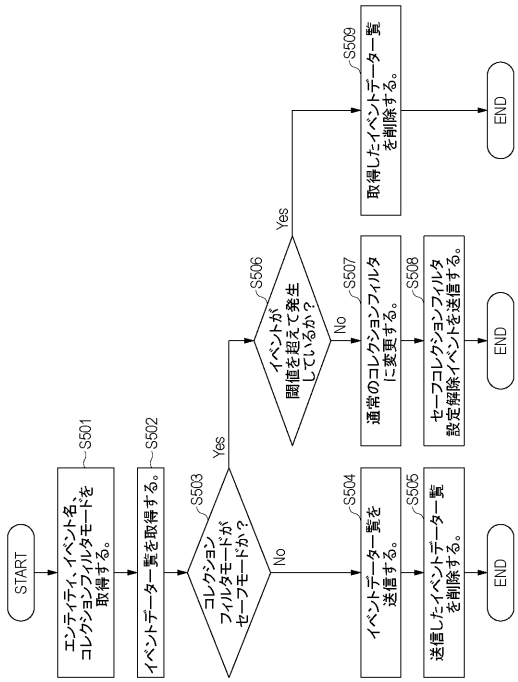
【図３】


(A) 102
クライアント端末
トークンプロバイダモジュール 301
データ送信モジュール 302
機器内モジュール 303
データ保持モジュール 304
コレクションフィルタ受信モジュール 305
(B) 112
第1の認証認可サーバ
第1の認証認可モジュール 311
第1のクライアント管理モジュール 312
(C) 113
データ収集サーバ
トークン取得モジュール 321
ストリーム処理モジュール 322
コレクションフィルタ送信モジュール 323
(D) 114
データ利用サーバ
トークン取得モジュール 331
メッセージ処理モジュール 332
(E) 122
第2の認証認可サーバ
第2の認証認可モジュール 341
第2のクライアント管理モジュール 342
(F) 123
ストリーム受信サーバ
ストリーム受信モジュール 351
(G) 124
メッセージサーバ
メッセージモジュール 361
送信管理モジュール 362
(H) 125
メッセージキューサーバ
メッセージキューモジュール 371

【図４】


ストリーム受信モジュール 351
第2の認証認可モジュール 341
トークン取得モジュール 321
第1の認証認可モジュール 311
トークンプロバイダモジュール 301
データ送信モジュール 302
S401 トークン発行要求
S402 トークン発行要求
S403 トークン発行要求
S404 トークンの検証
S405 トークン発行要求
S406 IDトークン発行要求
S407 トークン発行要求
S408 データ送信

【図５】


START
S501
エンティティ、イベント名、コレクションフィルタモードを取得する。
S502
イベントデーター覧を取得する。
S503
コレクション フィルタモードが セーフモードか？
Yes
No
S504
イベントデーター覧を送信する。
S505
送信したイベントデーター覧を削除する。
END
S506
イベントが 閾値を超えて発生 しているか？
Yes
No
S507
通常のコレクションフィルタに変更する。
S508
セーフコレクションフィルタ設定解除イベントを送信する。
END
S509
取得したイベントデーター覧を削除する。
END

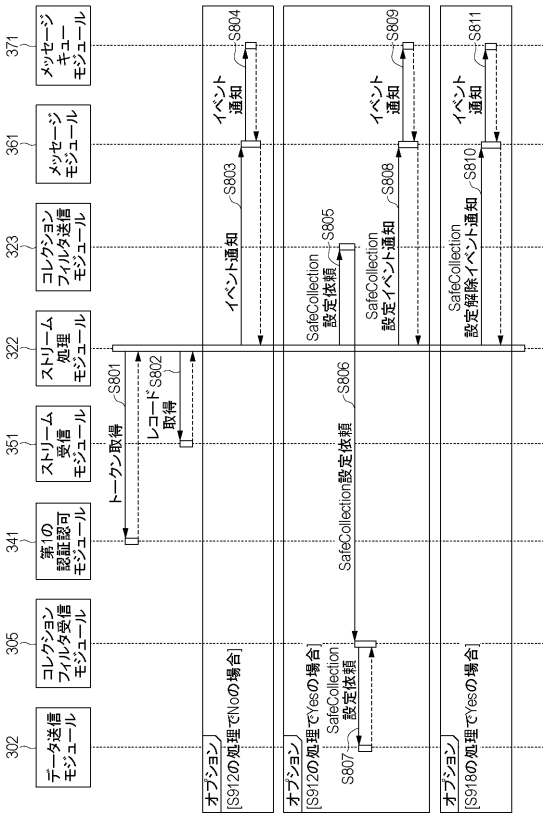
【図６】


(A)
601
{
"collections":[
{
"entity":"HDD",
602
"eventName":"sensorError",
603
604
"fire":"10min",
"collectionFilterMode":"normal"
605
}
{
"entity":"Print",
"eventName":"jobStart",
"fire":"1min",
"collectionFilterMode":"normal"
}
]
}
(B)
601
{
"collections":[
{
"entity":"HDD",
602
"eventName":"sensorError",
603
604
"fire":"60min",
"collectionFilterMode":"safe",
605
606
"threshold":5
}
{
"entity":"Print",
"eventName":"jobStart",
"fire":"1min",
"collectionFilterMode":"normal"
}
]
}

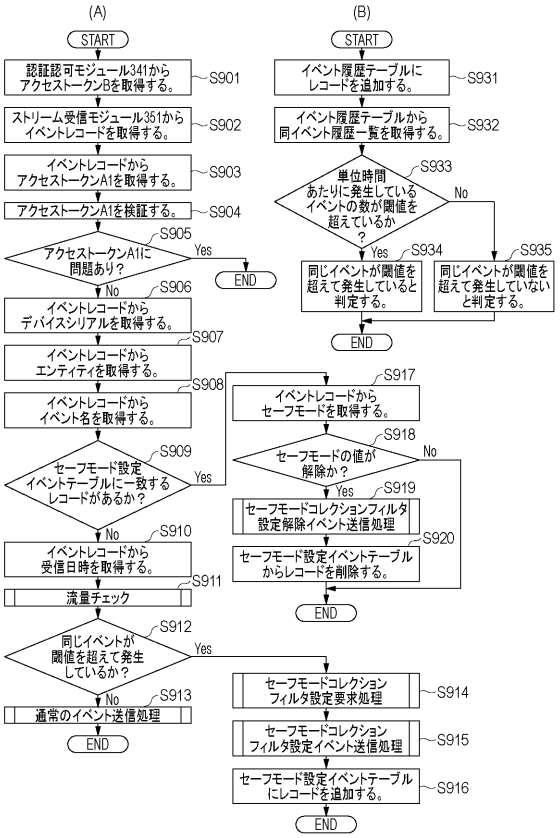
【図７】


(A)
701
{
"event":{
"id":"1235465423",
702
"deviceSerial":"AAA12345",
703
704
"entity":"HDD",
"eventName":"sensorError",
705
"occurDateTime":2018/08/31 00:00:01
706
}
}
(B)
701
{
"event":{
"id":"1235465423",
702
"deviceSerial":"AAA12345",
703
704
"entity":"HDD",
"eventName":"sensorError",
705
"occurDateTime":2018/08/31 00:00:01,
706
707
"safeMode":"release"
}
}

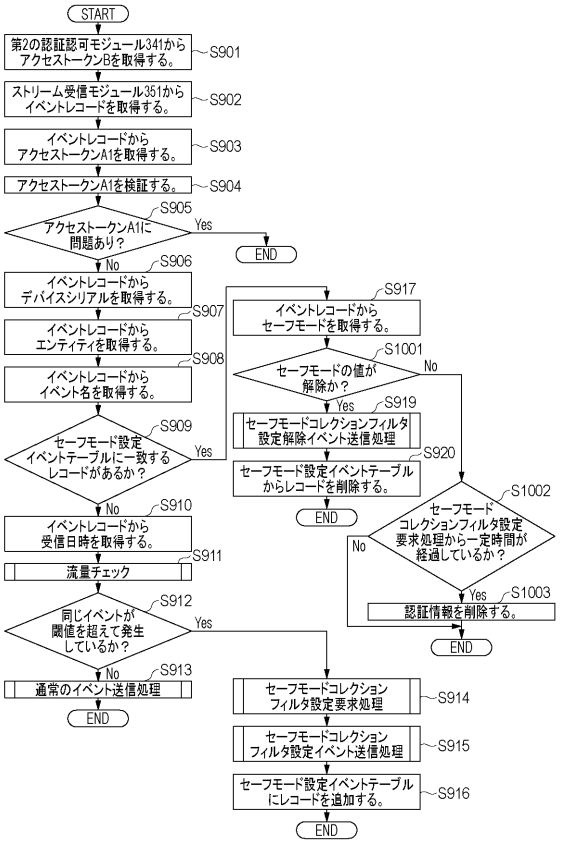
【図８】


371
メッセージ キュー モジュール
361
メッセージ モジュール
323
コレクション フィルタ送信 モジュール
322
ストリーム 処理 モジュール
351
ストリーム 受信 モジュール
341
第1の 認証認可 モジュール
305
コレクション フィルタ受信 モジュール
302
データ送信 モジュール
S804
イベント 通知
S803
イベント通知
S801
トークン取得
S802
レコード 取得
オプション
[S912の処理でNoの場合]
S809
イベント 通知
S808
SafeCollection 設定イベント通知
S805
SafeCollection 設定依頼
S806
SafeCollection設定依頼
S807
SafeCollection 設定依頼
オプション
[S912の処理でYesの場合]
S811
イベント 通知
S810
SafeCollection 設定解除イベント通知
オプション
[S918の処理でYesの場合]

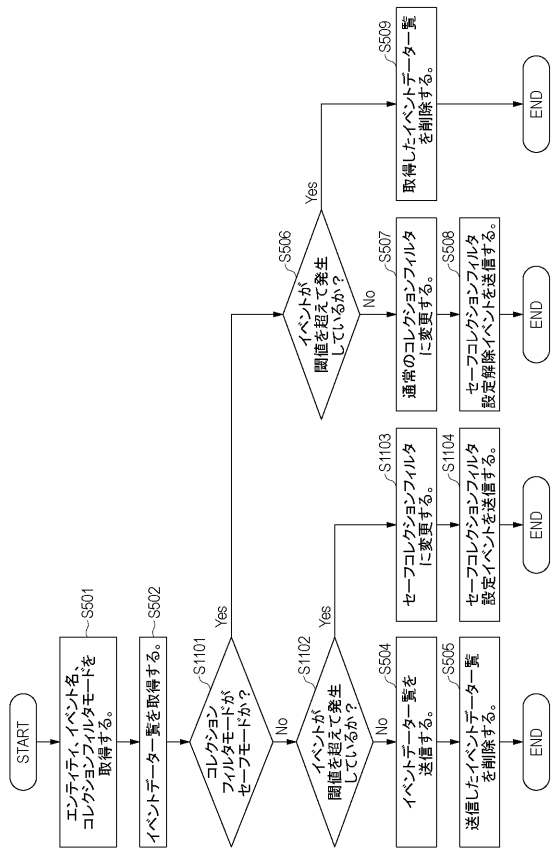
【図９】




【図１０】




【図１１】




【図１２】