

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7601112号
(P7601112)

(45)発行日 令和6年12月17日(2024.12.17)

(24)登録日 令和6年12月9日(2024.12.9)

(51)国際特許分類

F I

H 0 4 L 67/566 (2022.01)

H 0 4 L 67/566

請求項の数 3 (全12頁)

(21)出願番号	特願2022-561742(P2022-561742)	(73)特許権者	000004226
(86)(22)出願日	令和2年11月10日(2020.11.10)		日本電信電話株式会社
(86)国際出願番号	PCT/JP2020/041987		東京都千代田区大手町一丁目 5 番 1 号
(87)国際公開番号	WO2022/102001	(74)代理人	110002147
(87)国際公開日	令和4年5月19日(2022.5.19)		弁理士法人酒井国際特許事務所
審査請求日	令和5年3月8日(2023.3.8)	(72)発明者	石崎 晃朗
			東京都千代田区大手町一丁目 5 番 1 号
			日本電信電話株式会社内
		(72)発明者	サンバト プリヤンカラ
			東京都千代田区大手町一丁目 5 番 1 号
			日本電信電話株式会社内
		審査官	小林 義晴

最終頁に続く

(54)【発明の名称】 データフロー制御装置、データフロー制御方法、及びデータフロー制御プログラム

(57)【特許請求の範囲】

【請求項 1】

データを処理するシステムにおいて稼働中の第 1 のデータフローのユーザにあらかじめ設定された属性情報を基に、前記第 1 のデータフローの共通化が許可されているか否かを確認する確認部と、

前記第 1 のデータフローと、入力された第 2 のデータフローで共通する第 1 の処理の結果をキューに保存し、前記第 2 のデータフローにおいて前記第 1 の処理の結果を利用する第 2 の処理が、前記キューからデータを取得するような第 3 のデータフローを、前記確認部によって前記第 1 のデータフローの共通化が許可されていることが確認された場合に算出する算出部と、

前記システムに対し、前記第 1 のデータフローを前記第 3 のデータフローに切り替えることを指示する指示部と、

を有することを特徴とするデータフロー制御装置。

【請求項 2】

データフロー制御装置によって実行されるデータフロー制御方法であって、

データを処理するシステムにおいて稼働中の第 1 のデータフローのユーザにあらかじめ設定された属性情報を基に、前記第 1 のデータフローの共通化が許可されているか否かを確認する確認工程と、

前記第 1 のデータフローと、入力された第 2 のデータフローで共通する第 1 の処理の結果をキューに保存し、前記第 2 のデータフローにおいて前記第 1 の処理の結果を利用する第

2 の処理が、前記キューからデータを取得するような第 3 のデータフローを、前記確認工程によって前記第 1 のデータフローの共通化が許可されていることが確認された場合に算出する算出工程と、

前記システムに対し、前記第 1 のデータフローを前記第 3 のデータフローに切り替えることを指示する指示工程と、

を含むことを特徴とするデータフロー制御方法。

【請求項 3】

コンピュータを、請求項 1 に記載のデータフロー制御装置として機能させるためのデータフロー制御プログラム。

【発明の詳細な説明】

10

【技術分野】

【0001】

本発明は、データフロー制御装置、データフロー制御方法、及びデータフロー制御プログラムに関する。

【背景技術】

【0002】

従来、データ処理を行うワークフローのメタデータを基に、データ転送に関する機能を複数の論理的なレイヤに分けて管理する「Kafon」と呼ばれる手法が知られている（例えば、非特許文献 1 を参照）。

【0003】

20

非特許文献 1 に記載の手法は、システムの各機能を、アプリケーションロジック、転送経路の管理、ノード間の伝送、ミドルウェア実装の 4 つのレイヤに分けて管理し、転送処理の共通化等によりデータフローの効率化を図るものである。

【先行技術文献】

【非特許文献】

【0004】

【文献】Cyber Agent, "大規模データ処理", <https://www.cyberagent.co.jp/techinfo/labo/tech/detail/id=23934>

【文献】福田 一郎、他、"サイバーエージェントのデータ活用のための R&D 体制と取り組み", 人工知能学会, 30 巻 3 号, pp.318-324. (https://jsai.ixsq.nii.ac.jp/ej/?action=repository_action_common_download&item_id=1892&item_no=1&attribute_id=22&file_no=1)

30

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、従来の手法には、システム全体でのデータフローの効率化が困難な場合があるという問題がある。例えば、非特許文献 1 に記載の手法では、タスクを割り当てる演算器の能力や演算器間のデータ転送等が考慮されておらず、システム全体で必ずしも性能が向上するとは限らない。

【課題を解決するための手段】

40

【0006】

上述した課題を解決し、目的を達成するために、データフロー制御装置は、データを処理するシステムにおいて稼働中の第 1 のデータフローと、入力された第 2 のデータフローと、の少なくとも一部を共通化して統合したデータフローのうち、前記システムにおいて稼働させた場合のリソースの使用量が所定の条件を満たす第 3 のデータフローを算出する算出部と、前記システムに対し、前記第 1 のデータフローを前記第 3 のデータフローに切り替えることを指示する指示部と、を有することを特徴とする。

【発明の効果】

【0007】

本発明によれば、システム全体でデータフローを効率化することができる。

50

【図面の簡単な説明】**【 0 0 0 8 】**

【図 1】図 1 は、第 1 の実施形態に係るデータフロー制御装置の構成例を示す図である。

【図 2】図 2 は、データフローの一例を示す模式図である。

【図 3】図 3 は、データフローの切り替えについて説明する図である。

【図 4】図 4 は、第 1 の実施形態に係るデータフロー制御装置の処理の流れを示すフローチャートである。

【図 5】図 5 は、データフロー制御プログラムを実行するコンピュータの一例を示す図である。

【発明を実施するための形態】

10

【 0 0 0 9 】

以下に、本願に係るデータフロー制御装置、データフロー制御方法、及びデータフロー制御プログラムの実施形態を図面に基づいて詳細に説明する。なお、本発明は、以下に説明する実施形態により限定されるものではない。

【 0 0 1 0 】**[第 1 の実施形態の構成]**

まず、図 1 を用いて、第 1 の実施形態に係るデータフロー制御装置の構成について説明する。図 1 は、第 1 の実施形態に係るデータフロー制御装置の構成の一例を示す図である。

【 0 0 1 1 】

図 1 に示すように、データフロー制御装置 1 0 は、データフローの入力を受け付け、データフローの変更指示をデータ処理システム 2 0 に対して出力する。ユーザは、データフローの追加、更新、削除等のためのデータフロー定義情報をデータフロー制御装置 1 0 に入力する。データフロー制御装置 1 0 は、インタフェース部 1 1、記憶部 1 2 及び制御部 1 3 を有する。なお、データ処理システム 2 0 は、データを処理するシステムの一例である。

20

【 0 0 1 2 】

インタフェース部 1 1 は、データの入出力及びデータの通信を行うためのインタフェースである。例えば、インタフェース部 1 1 は、キーボード及びマウス等の入力装置からデータの入力を受け付ける。また、例えば、インタフェース部 1 1 は、ディスプレイ及びスピーカ等の出力装置にデータを出力する。また、例えば、インタフェース部 1 1 は NIC (Network Interface Card) であってもよい。

30

【 0 0 1 3 】

記憶部 1 2 は、HDD (Hard Disk Drive)、SSD (Solid State Drive)、光ディスク等の記憶装置である。なお、記憶部 1 2 は、RAM (Random Access Memory)、フラッシュメモリ、NVRAM (Non Volatile Static Random Access Memory) 等のデータを書き換え可能な半導体メモリであってもよい。記憶部 1 2 は、データフロー制御装置 1 0 で実行される OS (Operating System) や各種プログラムを記憶する。また、記憶部 1 2 は、フロー情報管理 DB 1 2 1、演算資源割当管理 DB 1 2 2 及び回線情報 DB 1 2 3 を記憶する。

【 0 0 1 4 】

40

フロー情報管理 DB 1 2 1 は、データ処理システム 2 0 で稼働中のデータフローに関する情報を保持する。フロー情報管理 DB 1 2 1 は、データフローに含まれる各処理のデータ取得元、処理方法及び処理順序等を保持する。また、フロー情報管理 DB 1 2 1 は、データ処理システム 2 0 が収集するデータ及びデータフローのユーザの属性に関する各種メタデータを保持する。

【 0 0 1 5 】

演算資源割当管理 DB 1 2 2 は、計算リソースごとの、実行可能な処理、及び各処理への割り当てがされているか否かといった情報を保持する。さらに、演算資源割当管理 DB 1 2 2 は、各ユーザに対する計算リソースの割り当てが許可されているか否か、すなわちアクセス制御に関する情報を保持する。

50

【 0 0 1 6 】

回線情報 DB 1 2 3 は、サーバ間、ユーザとサーバ間、データ収集元とサーバ間、データセンタ間のネットワークの遅延や帯域に関する情報を保持する。なお、ここでのサーバは、例えばデータ処理システム 2 0 を構成する複数のサーバである。

【 0 0 1 7 】

制御部 1 3 は、データフロー制御装置 1 0 全体を制御する。制御部 1 3 は、例えば、CPU (Central Processing Unit)、MPU (Micro Processing Unit)、GPU (Graphics Processing Unit) 等の電子回路や、ASIC (Application Specific Integrated Circuit)、FPGA (Field Programmable Gate Array) 等の集積回路である。また、制御部 1 3 は、各種の処理手順を規定したプログラムや制御データを格納するための内部メモリを有し、内部メモリを用いて各処理を実行する。また、制御部 1 3 は、各種のプログラムが動作することにより各種の処理部として機能する。例えば、制御部 1 3 は、確認部 1 3 1、抽出部 1 3 2、算出部 1 3 3、指示部 1 3 4 及び更新部 1 3 5 を有する。

10

【 0 0 1 8 】

確認部 1 3 1 は、フロー情報管理 DB 1 2 1 を参照し、入力されたデータフローに対応するユーザの属性情報が、データフローの共有を許可するものであるか否かを確認する。さらに、確認部 1 3 1 は、入力されたデータフローと稼働中のデータフローとの間に重複部分があるか否かを確認する。

【 0 0 1 9 】

例えば、フロー情報管理 DB 1 2 1 に、優先利用の権限を持つユーザが保有するデータフローによって、演算器及びネットワーク資源等のリソースが利用されていることが示されている場合、当該リソースは当該ユーザによって占有される。そのため、確認部 1 3 1 は、優先利用の権限を持つユーザが保有するデータフローと入力されたデータフローとの統合は許可されていないと判断する。

20

【 0 0 2 0 】

ここで、図 2 を用いて、データフローについて説明する。図 2 は、データフローの一例を示す模式図である。図 2 の既存のデータフローは、データ処理システム 2 0 で稼働中のデータフローである。また、追加のデータフローは、データフロー制御装置 1 0 にユーザによって入力されたデータフローである。

30

【 0 0 2 1 】

図 2 に示すように、既存のデータフローと、追加のデータフローとで、「Camera → 処理 P 1」の部分が重複している。このため、図 2 の例では、確認部 1 3 1 は、入力されたデータフローと稼働中のデータフローとの間に重複部分があることを確認する。

【 0 0 2 2 】

抽出部 1 3 2 は、既存のデータフローのうち、共有可能なデータフローを抽出する。例えば、抽出部 1 3 2 は、ユーザの属性情報によって共有が許可されており、かつ重複部分があるデータフローを抽出する。

【 0 0 2 3 】

算出部 1 3 3 は、抽出部 1 3 2 によって抽出されたデータフローと、入力されたデータフローと、の少なくとも一部を共通化して統合したデータフローのうちの、データ処理システム 2 0 において稼働させた場合のリソースの使用量が所定の条件を満たすような新たなデータフローを算出する。

40

【 0 0 2 4 】

ここで、抽出部 1 3 2 によって抽出されたデータフローは、データを処理するシステムにおいて稼働中の第 1 のデータフローの一例である。また、入力されたデータフローは第 2 のデータフローの一例である。また、新たなデータフローは第 3 のデータフローの一例である。

【 0 0 2 5 】

まず、算出部 1 3 3 は、抽出部 1 3 2 によって抽出されたデータフローのそれぞれにつ

50

いて、各データフローを実現可能な空き計算リソース及びネットワークリソースの組み合わせを作成する。このとき、算出部 133 は、演算資源割当管理 DB 122 及び回線情報 DB 123 を参照して組み合わせを作成することができる。

【0026】

さらに、算出部 133 は、作成した各組み合わせについてリソースの使用量、すなわちコストを計算し、例えばコストが最も小さい組み合わせを選定する。算出部 133 は、各組み合わせの演算資源間、データ提供元と演算資源間、及びデータ出力先と演算資源間のデータ転送遅延時間、ネットワーク NW 帯域情報を基にコストを計算する。

【0027】

演算資源は、例えばデータ処理システム 20 を構成する情報処理装置の演算装置等である。また、データ提供元は、例えばカメラ及びセンサ等である。また、データ出力先は、例えば他の情報処理装置及びディスプレイ等の出力装置である。

10

【0028】

また、前述のように、確認部 131 は、稼働中のデータフローのユーザにあらかじめ設定された属性情報を基に、稼働中のデータフローの共通化が許可されているか否かを確認する。そして、算出部 133 は、確認部 131 によって稼働中のデータフローの共通化が許可されていることが確認された場合に、共通化されたデータフローを算出する。

【0029】

指示部 134 は、システムに対し、抽出部 132 によって抽出されたデータフローを、算出部 133 によって算出された新たなデータフローに切り替えることを指示する。

20

【0030】

図 3 は、データフローの切り替えについて説明する図である。図 3 に示すように、切り替え前のデータフローでは、ユーザ U1 がアプリケーション AP1 を実行した場合に、演算器 a によって処理 P1 が実行されていた。

【0031】

ここで、ユーザ U2 によるアプリケーション AP1 の実行、及びユーザ U2 によるアプリケーション AP1 の実行に関するデータフローが追加されたものとする。切り替え前のデータフローと、追加された 2 つのデータフローとでは、処理 P1 を実行する部分が重複している。このため、算出部 133 は、切り替え前のデータフローと追加されたデータフローを統合した「Camera→処理 P1 (演算器 b)→ユーザ U1 AP1」というデータフローを算出する。

30

【0032】

そして、算出部 133 によって「Camera→処理 P1 (演算器 b)→ユーザ U1 AP1」というデータフローが選定されると、指示部 134 は、既存のデータフロー「Camera→処理 P1 (演算器 a)→ユーザ U1 AP1」を、選定されたデータフロー「Camera→処理 P1 (演算器 b)→ユーザ U1 AP1」に切り替えることをデータ処理システム 20 に指示する。

【0033】

また、切り替え前後では、処理 P1 を実行する演算器が変化している。例えば、演算器 b は、処理 P2 及び処理 P3 との間に設けられたキューに、処理 P1 の実行結果を保存する。

40

【0034】

このように、算出部 133 は、既存のデータフローと入力されたデータフローで共通する処理 P1 の結果をキューに保存し、入力されたデータフローにおいて処理 P1 の結果を利用する処理 P2 及び処理 P3 が、キューからデータを取得するような統合済みのデータフローを算出することができる。言い換えると、演算器 c 及び演算器 d は、それぞれ処理 P2 及び処理 P3 を実行する過程で、キューから処理 P1 の実行結果を取得する。

【0035】

この場合、データ処理システム 20 は、処理 P2 及び処理 P3 の実行の際に、処理 P1 を実行することなく、キューから実行結果を取得することができる。なお、処理 P1 の実

50

行結果は、例えば検索によって得られたデータの一覧等である。

【 0 0 3 6 】

なお、算出部 1 3 3 は、処理 P 1 の実行主体を演算器 a から演算器 b に変更した結果、全体のコストが増大すると判断した場合、図 3 のようなデータフローの切り替えを選定しないようにすることができる。

【 0 0 3 7 】

更新部 1 3 5 は、データフローの新旧切り替えに伴い、各 D B の情報を更新する。ここで、切り替え前のデータフローを旧データフローと呼ぶ。また、切り替え後のデータフローを新データフローと呼ぶ。

【 0 0 3 8 】

まず、更新部 1 3 5 は、新旧データフローの切り替え前に、新データフローのための計算リソースを確保し、演算資源割当管理 D B 1 2 2 を更新する。そして、更新部 1 3 5 は、新旧データフローの切り替え後に、フロー情報管理 D B 1 2 1 上に新データフローの情報を追加し、旧データフローの情報を更新する。

【 0 0 3 9 】

そして、更新部 1 3 5 は、旧データフローで利用していた計算リソースを解放し演算資源割当管理 D B 1 2 2 を更新する。さらに、更新部 1 3 5 は、旧データフローのフロー情報管理 D B 1 2 1 を更新する。

【 0 0 4 0 】

[第 1 の実施形態の処理]

図 4 は、第 1 の実施形態に係るデータフロー制御装置の処理の流れを示すフローチャートである。まず、データフロー制御装置 1 0 は、データフロー情報の入力を受け付ける（ステップ S 1 0 1 ）。

【 0 0 4 1 】

次に、データフロー制御装置 1 0 は、ユーザ情報からデータフローを共有するか確認する（ステップ S 1 0 2 ）。データフローを共有しない場合（ステップ S 1 0 3、N o ）、データフロー制御装置 1 0 はステップ S 1 0 7 に進む。一方、データフローを共有する場合（ステップ S 1 0 3、Y e s ）、データフロー制御装置 1 0 はステップ S 1 0 4 に進む。

【 0 0 4 2 】

続いて、データフロー制御装置 1 0 は、入力されたデータフローに既存データフローと重複部分があるか否かを判定する（ステップ S 1 0 4 ）。入力されたデータフローに既存データフローと重複部分がない場合（ステップ S 1 0 4、N o ）、データフロー制御装置 1 0 はステップ S 1 0 7 に進む。一方、入力されたデータフローに既存データフローと重複部分がある場合（ステップ S 1 0 4、Y e s ）、データフロー制御装置 1 0 はステップ S 1 0 5 に進む。

【 0 0 4 3 】

データフロー制御装置 1 0 は、共有可能な既存データフローを抽出する（ステップ S 1 0 5 ）。そして、データフロー制御装置 1 0 は、入力されたデータフローと抽出したデータフローとを統合したデータフローを算出する（ステップ S 1 0 6 ）。

【 0 0 4 4 】

データフロー制御装置 1 0 は、算出したデータフローを実現可能な空き計算リソース及び N W（ネットワーク）リソースの組み合わせを作成する（ステップ S 1 0 7 ）。そして、データフロー制御装置 1 0 は、入力から出力までのコストを基に組み合わせを選定する（ステップ S 1 0 8 ）。

【 0 0 4 5 】

ここで、データフロー制御装置 1 0 は、計算リソースの確保をし、演算資源割当管理 D B 1 2 2 を更新する（ステップ S 1 0 9 ）。そして、データフロー制御装置 1 0 は、データフローの新旧切り替えをデータ処理システム 2 0 に指示する（ステップ S 1 1 0 ）。

【 0 0 4 6 】

また、データフロー制御装置 1 0 は、フロー情報管理 D B 1 2 1 上に新データフローの

10

20

30

40

50

情報を追加し、旧データフローの情報を更新する（ステップ S 1 1 1）。さらに、データフロー制御装置 1 0 は、旧データフローで利用していた計算リソースを解放し演算資源割当管理 DB 1 2 2 を更新する（ステップ S 1 1 2）。そして、データフロー制御装置 1 0 は、旧データフローのフロー情報管理 DB 1 2 1 を更新する（ステップ S 1 1 3）。

【 0 0 4 7 】

[第 1 の実施形態の効果]

これまで説明してきたように、データフロー制御装置 1 0 は、データを処理するシステムにおいて稼働中の第 1 のデータフローと、入力された第 2 のデータフローと、の一部を共通化して統合したデータフローのうち、システムにおいて稼働させた場合のリソースの使用量が所定の条件を満たすような第 3 のデータフローを算出する。データフロー制御装置 1 0 は、システムに対し、第 1 のデータフローを第 3 のデータフローに切り替えることを指示する。このように、データフロー制御装置 1 0 は、新旧のデータフローの一部を共通化するだけでなく、システムのリソース状況を考慮した上で切り替えを行う。その結果、本実施形態によれば、システム全体でデータフローを効率化することができる。また、本実施形態によれば、データフローのプラグブルな処理の入れ替えが可能になる。

10

【 0 0 4 8 】

データフロー制御装置 1 0 は、第 1 のデータフローと第 2 のデータフローで共通する第 1 の処理の結果をキューに保存し、第 2 のデータフローにおいて第 1 の処理の結果を利用する第 2 の処理が、キューからデータを取得するように第 3 のデータフローを算出する。この結果、本実施形態によれば、システム全体の処理効率を向上させることができる。

20

【 0 0 4 9 】

データフロー制御装置 1 0 は、第 1 のデータフローのユーザにあらかじめ設定された属性情報を基に、第 1 のデータフローの共通化が許可されているか否かを確認する。データフロー制御装置 1 0 は、確認部 1 3 1 によって第 1 のデータフローの共通化が許可されていることが確認された場合に、第 3 のデータフローを算出する。データの出力先が複数のユーザであるマルチテナントのシステムが存在する。そのようなシステムの場合、ユーザの属性によっては、データフローの共通化が許可されていない場合がある。本実施形態では、マルチテナントを考慮した上でシステム全体の処理の効率化を図ることができる。

【 0 0 5 0 】

[システム構成等]

また、図示した各装置の各構成要素は機能概念的なものであり、必ずしも物理的に図示のように構成されていることを要しない。すなわち、各装置の分散及び統合の具体的形態は図示のものに限られず、その全部又は一部を、各種の負荷や使用状況等に応じて、任意の単位で機能的又は物理的に分散又は統合して構成することができる。さらに、各装置にて行われる各処理機能は、その全部又は任意の一部が、CPU (Central Processing Unit) 及び当該 CPU にて解析実行されるプログラムにて実現され、あるいは、ワイヤードロジックによるハードウェアとして実現され得る。なお、プログラムは、CPU だけでなく、GPU 等の他のプロセッサによって実行されてもよい。

30

【 0 0 5 1 】

また、本実施形態において説明した各処理のうち、自動的に行われるものとして説明した処理の全部又は一部を手動的に行うこともでき、あるいは、手動的に行われるものとして説明した処理の全部又は一部を公知の方法で自動的に行うこともできる。この他、上記文書中や図面中で示した処理手順、制御手順、具体的名称、各種のデータやパラメータを含む情報については、特記する場合を除いて任意に変更することができる。

40

【 0 0 5 2 】

[プログラム]

一実施形態として、データフロー制御装置 1 0 は、パッケージソフトウェアやオンラインソフトウェアとして上記のデータフロー制御処理を実行するデータフロー制御プログラムを所望のコンピュータにインストールさせることによって実装できる。例えば、上記のデータフロー制御プログラムを情報処理装置に実行させることにより、情報処理装置をデ

50

ータフロー制御装置 10 として機能させることができる。ここで言う情報処理装置には、デスクトップ型又はノート型のパーソナルコンピュータが含まれる。また、その他にも、情報処理装置にはスマートフォン、携帯電話機や P H S (Personal Handyphone System) 等の移動体通信端末、さらには、P D A (Personal Digital Assistant) 等のスレート端末等がその範疇に含まれる。

【0053】

また、データフロー制御装置 10 は、ユーザが使用する端末装置をクライアントとし、当該クライアントに上記のデータフロー制御処理に関するサービスを提供するデータフロー制御サーバ装置として実装することもできる。例えば、データフロー制御サーバ装置は、データフローを入力とし、データフローの変更指示を出力とするデータフロー制御サービスを提供するサーバ装置として実装される。この場合、データフロー制御サーバ装置は、Webサーバとして実装することとしてもよいし、アウトソーシングによって上記のデータフロー制御処理に関するサービスを提供するクラウドとして実装することとしてもかまわない。

10

【0054】

図 5 は、データフロー制御プログラムを実行するコンピュータの一例を示す図である。コンピュータ 1000 は、例えば、メモリ 1010、CPU 1020 を有する。また、コンピュータ 1000 は、ハードディスクドライブインタフェース 1030、ディスクドライブインタフェース 1040、シリアルポートインタフェース 1050、ビデオアダプタ 1060、ネットワークインタフェース 1070 を有する。これらの各部は、バス 1080 によって接続される。

20

【0055】

メモリ 1010 は、ROM (Read Only Memory) 1011 及び RAM (Random Access Memory) 1012 を含む。ROM 1011 は、例えば、BIOS (Basic Input Output System) 等のブートプログラムを記憶する。ハードディスクドライブインタフェース 1030 は、ハードディスクドライブ 1090 に接続される。ディスクドライブインタフェース 1040 は、ディスクドライブ 1100 に接続される。例えば磁気ディスクや光ディスク等の着脱可能な記憶媒体が、ディスクドライブ 1100 に挿入される。シリアルポートインタフェース 1050 は、例えばマウス 1110、キーボード 1120 に接続される。ビデオアダプタ 1060 は、例えばディスプレイ 1130 に接続される。

30

【0056】

ハードディスクドライブ 1090 は、例えば、OS 1091、アプリケーションプログラム 1092、プログラムモジュール 1093、プログラムデータ 1094 を記憶する。すなわち、データフロー制御装置 10 の各処理を規定するプログラムは、コンピュータにより実行可能なコードが記述されたプログラムモジュール 1093 として実装される。プログラムモジュール 1093 は、例えばハードディスクドライブ 1090 に記憶される。例えば、データフロー制御装置 10 における機能構成と同様の処理を実行するためのプログラムモジュール 1093 が、ハードディスクドライブ 1090 に記憶される。なお、ハードディスクドライブ 1090 は、SSD (Solid State Drive) により代替されてもよい。

40

【0057】

また、上述した実施形態の処理で用いられる設定データは、プログラムデータ 1094 として、例えばメモリ 1010 やハードディスクドライブ 1090 に記憶される。そして、CPU 1020 は、メモリ 1010 やハードディスクドライブ 1090 に記憶されたプログラムモジュール 1093 やプログラムデータ 1094 を必要に応じて RAM 1012 に読み出して、上述した実施形態の処理を実行する。

【0058】

なお、プログラムモジュール 1093 やプログラムデータ 1094 は、ハードディスクドライブ 1090 に記憶される場合に限らず、例えば着脱可能な記憶媒体に記憶され、ディスクドライブ 1100 等を介して CPU 1020 によって読み出されてもよい。あるいは

50

は、プログラムモジュール 1 0 9 3 及びプログラムデータ 1 0 9 4 は、ネットワーク（L A N（Local Area Network）、W A N（Wide Area Network）等）を介して接続された他のコンピュータに記憶されてもよい。そして、プログラムモジュール 1 0 9 3 及びプログラムデータ 1 0 9 4 は、他のコンピュータから、ネットワークインタフェース 1 0 7 0 を介して C P U 1 0 2 0 によって読み出されてもよい。

【符号の説明】

【 0 0 5 9 】

1 0 データフロー制御装置

1 1 インタフェース部

1 2 記憶部

1 3 制御部

2 0 データ処理システム

1 2 1 フロー情報管理 D B

1 2 2 演算資源割当管理 D B

1 2 3 回線情報 D B

1 3 1 確認部

1 3 2 抽出部

1 3 3 算出部

1 3 4 指示部

1 3 5 更新部

10

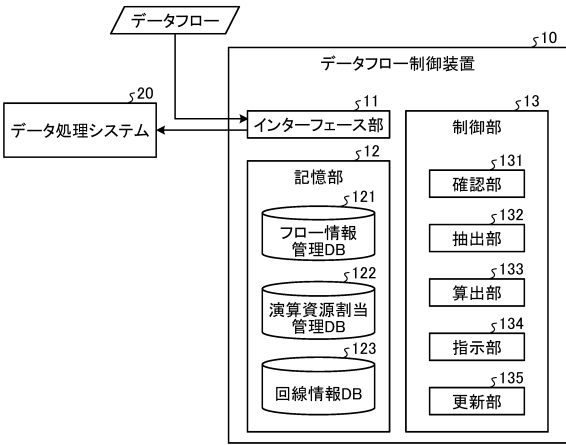
20

30

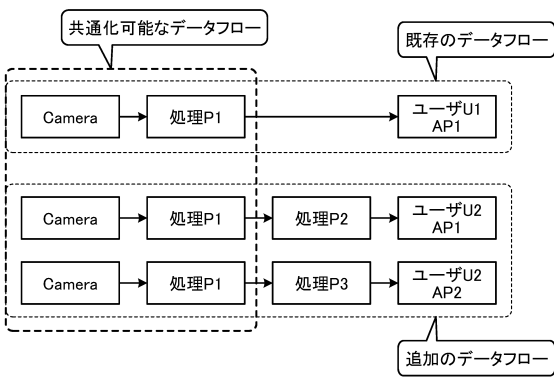
40

50

【 図 面 】
【 図 1 】



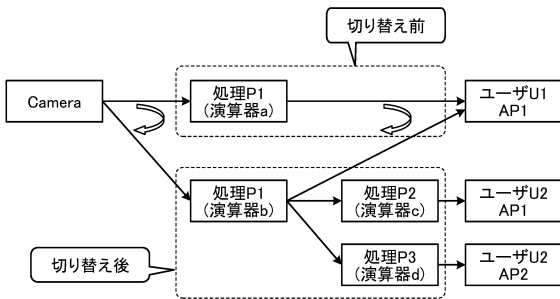
【 図 2 】



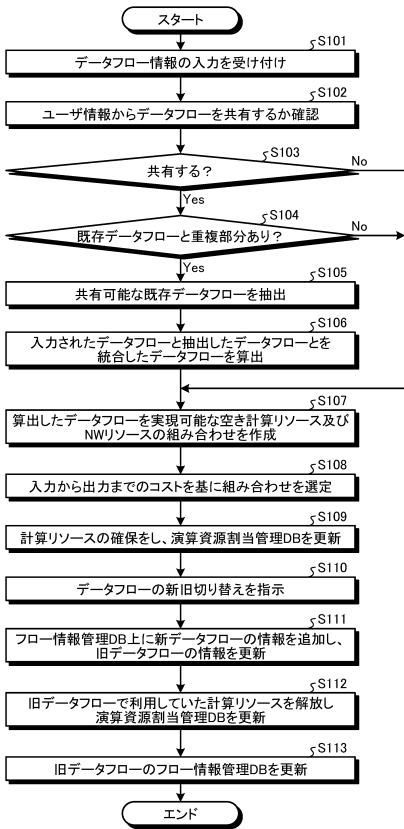
10

20

【 図 3 】



【 図 4 】

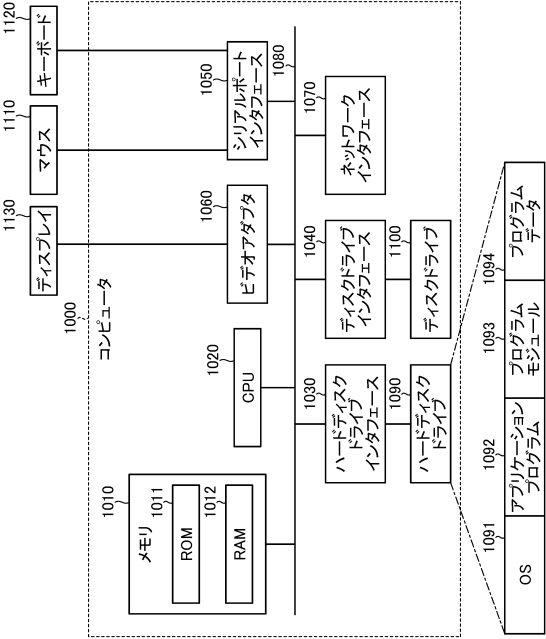


30

40

50

【図 5】



10

20

30

40

50

フロントページの続き

- (56)参考文献 国際公開第 2 0 2 0 / 1 1 5 8 8 8 (W O , A 1)
特開 2 0 1 7 - 1 1 1 5 3 3 (J P , A)
特開 2 0 0 8 - 1 3 0 0 4 8 (J P , A)
- (58)調査した分野 (Int.Cl., D B 名)
H 0 4 L 6 7 / 5 6 6