

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7553173号
(P7553173)

(45)発行日 令和6年9月18日(2024.9.18)

(24)登録日 令和6年9月9日(2024.9.9)

(51)国際特許分類	F I
G 0 6 F 11/34 (2006.01)	G 0 6 F 11/34 1 4 2
G 0 6 F 11/30 (2006.01)	G 0 6 F 11/30 1 4 0 A
G 0 6 F 9/50 (2006.01)	G 0 6 F 9/50 1 5 0 D

請求項の数 15 (全40頁)

(21)出願番号	特願2021-6670(P2021-6670)	(73)特許権者	524132520 日立ヴァンタラ株式会社 神奈川県横浜市戸塚区吉田町292番地
(22)出願日	令和3年1月19日(2021.1.19)	(74)代理人	110000279 弁理士法人ウィルフォート国際特許事務所
(65)公開番号	特開2022-110929(P2022-110929 A)	(72)発明者	仲野 香緒里 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
(43)公開日	令和4年7月29日(2022.7.29)	(72)発明者	林 真一 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
審査請求日	令和5年5月23日(2023.5.23)	(72)発明者	金子 聡 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内

最終頁に続く

(54)【発明の名称】 管理計算機、管理システム、及び管理プログラム

(57)【特許請求の範囲】

【請求項1】

ジョブを実行するジョブ実行サーバと、前記ジョブ実行サーバとネットワークを介して接続され、前記ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤を管理する管理計算機であって、

前記管理計算機は、記憶デバイスと、前記記憶デバイスに接続されたプロセッサとを備え、

前記記憶デバイスは、前記データ処理基盤の前記ジョブ実行サーバと前記ストレージ装置との間の通信に関わる構成要素の最大のリソース量の情報である最大リソース量情報と、前記データ処理基盤の前記ストレージ装置のデータへのパスの情報であるパス情報と、前記データ処理基盤の前記構成要素の負荷の情報である負荷情報とを記憶し、

前記プロセッサは、

前記最大リソース量情報と、前記パス情報と、前記負荷情報とに基づいて、所定のジョブの実行に関わる、前記ジョブ実行サーバから前記ストレージ装置のデータへのパスを構成する構成要素の空きリソース量を計算し、

前記空きリソース量に基づいて、前記ジョブ実行サーバにおける前記所定のジョブの実行時における前記ジョブを使用する並列実行可能な処理単位に対する並列して実行可能な最大の数である並列可能数を決定する

管理計算機。

【請求項2】

前記記憶デバイスは、ジョブで並列可能な処理単位の1単位あたりのデータ容量に関する情報であるデータ容量情報を記憶し、

前記プロセッサは、前記データ容量情報に基づいて、前記ジョブの並列可能な処理単位のデータのデータ容量を特定し、特定した前記データ容量に基づいて並行可能な処理単位の1単位の実行に必要なリソース量である要求リソース量を計算し、前記空きリソース量と、前記要求リソース量とに基づいて前記並列可能数を決定する

請求項1に記載の管理計算機。

【請求項3】

前記記憶デバイスは、データ転送に係る応答時間の情報である応答時間情報を更に記憶し、

前記プロセッサは、前記応答時間情報と前記データ容量とに基づいて、前記要求リソース量を計算する

請求項2に記載の管理計算機。

【請求項4】

前記記憶デバイスは、ジョブに含まれるプロセスの種別毎の情報であるプロセス種別情報を記憶し、

前記プロセッサは、前記プロセス種別情報と、前記所定のジョブに含まれる前記プロセスの種別とに基づいて、前記要求リソース量を計算する

請求項2に記載の管理計算機。

【請求項5】

前記プロセッサは、前記決定した並列可能数を表示させる

請求項1に記載の管理計算機。

【請求項6】

前記プロセッサは、

前記並列可能数に基づいて、前記ジョブ実行サーバにおけるジョブの並列可能な処理単位の実行時における並列数を設定する

請求項1に記載の管理計算機。

【請求項7】

前記プロセッサは、

前記並列可能数に基づいて、前記所定のジョブの実行に必要な前記ジョブ実行サーバの性能を決定する

請求項1に記載の管理計算機。

【請求項8】

前記プロセッサは、

前記所定のジョブで使用する前記データ処理基盤の前記構成要素の負荷を予測し、

前記予測した負荷に基づいて前記並列可能数を決定する

請求項1に記載の管理計算機。

【請求項9】

前記データ処理基盤の一部の構成要素は、自動的に台数を変更可能なオートスケール設定が適用されており、

前記プロセッサは、前記オートスケール設定に基づいて、前記空きリソース量を計算する

請求項1に記載の管理計算機。

【請求項10】

前記記憶デバイスは、前記データ処理基盤で実行される他のジョブの開始時刻と並列数とを含む他ジョブ情報を記憶し、

前記プロセッサは、

前記他ジョブ情報に基づいて、前記所定のジョブの開始時刻と前記並列可能数とを決定する

請求項1に記載の管理計算機。

【請求項11】

10

20

30

40

50

前記プロセッサは、

前記データ処理基盤で実行される各ジョブの完了許容期限に基づいて、前記完了許容期限を満たす最小の並列数を算出し、

前記所定のジョブの実行に必要な並列数が得られない場合に、他のジョブの中から並列数を減らすジョブを決定する

請求項 1 に記載の管理計算機。

【請求項 1 2】

前記プロセッサは、

前記データ処理基盤で実行される各ジョブの完了許容期限に基づいて、前記完了許容期限を満たす最小の要求リソース量を算出し、

前記所定のジョブの実行に必要な並列数が得られない場合に、他のジョブの中から割り当てるリソース量を減らすジョブを決定する

請求項 1 に記載の管理計算機。

【請求項 1 3】

前記所定のジョブは、複数の並列可能な処理単位に対する処理を含むプロセスを複数含み、

前記プロセッサは、

前記プロセスごとに並列可能数を決定し、

前記プロセスごとの並列可能数を表示させる

請求項 1 に記載の管理計算機。

【請求項 1 4】

ジョブを実行するジョブ実行サーバと、前記ジョブ実行サーバとネットワークを介して接続され、ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤における前記ジョブ実行サーバ又はストレージ装置との少なくとも一方と、前記データ処理基盤を管理する管理計算機とを備える管理システムであって、

前記管理計算機は、前記データ処理基盤の前記ジョブ実行サーバと前記ストレージ装置との間の通信に関わる構成要素の最大のリソース量の情報である最大リソース量情報と、前記データ処理基盤の前記ジョブ実行サーバから前記ストレージ装置のデータへのパスの情報であるパス情報と、前記データ処理基盤の前記構成要素の負荷の情報である負荷情報とを記憶し、

前記管理計算機は、

前記最大リソース量情報と、前記パス情報と、前記負荷情報とに基づいて、所定のジョブの実行に関わる前記ストレージ装置のデータへのパスを構成する構成要素の空きリソース量を計算し、

前記空きリソース量に基づいて、前記ジョブ実行サーバにおける前記所定のジョブの実行時における前記ジョブで使用する並列実行可能な処理単位に対する並列して実行可能な最大の数である並列可能数を決定する

管理システム。

【請求項 1 5】

ジョブを実行するジョブ実行サーバと、前記ジョブ実行サーバとネットワークを介して接続され、ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤を管理するコンピュータに実行させる管理プログラムであって、

前記コンピュータに、

前記データ処理基盤の前記ジョブ実行サーバと前記ストレージ装置との間の通信に関わる構成要素の最大のリソース量の情報である最大リソース量情報と、前記データ処理基盤の前記ストレージ装置のデータへのパスの情報であるパス情報と、前記データ処理基盤の前記構成要素の負荷の情報である負荷情報とに基づいて、所定のジョブの実行に関わる前記ストレージ装置のデータへのパスを構成する構成要素の空きリソース量を計算させ、

前記空きリソース量に基づいて、前記ジョブ実行サーバにおける前記所定のジョブの実行時における前記ジョブで使用する並列実行可能な処理単位に対する並列して実行可能な

10

20

30

40

50

最大の数である並列可能数を決定させる管理プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ジョブを実行するジョブ実行サーバと、ジョブ実行サーバとネットワークを介して接続され、ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤を管理する技術に関する。

【背景技術】

【0002】

近年、IT (Information Technology) の分野では、一企業が保有するデータセンタ (オンプレミス) でITシステムを構築するのではなく、パブリッククラウドを利用するケースが多い。一般的なパブリッククラウドは、データセンタ事業者がサーバ、ディスク (ストレージとも呼ぶ)、ネットワークなどの計算リソースをプール化し、仮想化技術などを用いて利用者ごとに分割して提供するサービスである。パブリッククラウドでは、利用した計算リソースの性能 (サーバのCPUコア数などの計算リソース量やディスクの種類などの品質) と利用時間に基づく従量課金が一般的である。これらの特徴から、大量のデータを比較的短期間に処理するデータ分析システムを構築する場合は、オンプレミスで構築するよりもパブリッククラウドを利用する方が費用を抑えることができる。

【0003】

また、近年は、特定の分析目的のためのシステムではなく、組織全体でデータを活用するために、企業内データや一般公開されたデータを集約し、適切にデータを抽出・加工・転送できるデータ分析基盤 (データ処理基盤の一例) をパブリッククラウドに構築する事例が増えている。

【0004】

一方で、データガバナンス等の観点から、ITシステムで処理するデータに関してはパブリッククラウドに置かず自社で管理したいというニーズがある。このようなニーズに対しては、データ分析基盤として、データを処理するコンピュータをパブリッククラウドに置く一方、データを格納したストレージ装置をオンプレミスに置き、これらをネットワークで接続するハイブリッドクラウド構成をとるケースが考えられる。

【0005】

データ分析基盤を利用するデータ分析者は、データ分析を開始する前に、集約されたデータ群 (以下、データレイクと呼ぶ) のデータカタログから分析目的に応じて利用するデータを選択し、必要なデータを抽出・加工・転送する処理 (以下、ETL処理と呼ぶ) を実行して、特定の分析目的用のデータ群として別の保存領域に保存する。このETL処理は、類似した複数のデータに対して同じ処理を行うため、これらの処理は、並列に実行可能であることが多い。例えば、処理を並列実行可能な場合においては、データ分析者は、できるだけ早くデータ処理を完了させるために、データを処理するコンピュータ (例えば、サーバ) の性能や数量、および、処理の並列数を大きく設定して処理を実行する場合がある。

【0006】

しかし、データ分析者はデータ分析基盤のシステム構成を把握していないため、データ処理用のコンピュータにデータを転送するためのサーバやネットワークやストレージ装置がどの程度のデータ転送量でボトルネックになるかを見積もることはできない。そのため、データ転送のためのシステムが処理性能のボトルネックとなり、コンピュータの性能、数量、および実行可能な処理の並列数を増やしても処理時間が短くならず、処理を実行するコンピュータをパブリッククラウドに置いている場合には、パブリッククラウドの従量課金制により、課金額が大きくなってしまふ。

【0007】

例えば、特許文献1には、サーバ装置に対して同時送信する処理要求の数 (サーバ装置

10

20

30

40

50

における並列処理数)を変えた各条件下で、条件ごとの応答時間を測定して、その処理効率を推定し、その推定結果に基づいてサーバ装置における最適な最大並列処理数を決定する技術が開示されている。

【先行技術文献】

【特許文献】

【0008】

【文献】特開2006-221516号公報

【発明の概要】

【発明が解決しようとする課題】

【0009】

しかし、データ分析基盤では、データを転送するために構築されたサーバ、ネットワーク、ストレージ装置の構成要素が複数のデータ処理と共有されるため、利用状況によってデータの最大転送速度は変動する。また、ハイブリッドクラウド構成のデータ分析基盤では、データを処理するコンピュータと、データとがネットワーク的に離れた場所に配置され、ネットワークは複数の企業で共有されるため、1つのデータを送受信するためにかかる時間は変動し、データ処理から要求されるデータ転送速度も変動する。そのため、同一のデータ処理内容であっても最適な並列処理数は固定的ではなく、特許文献1の技術では、最適な並列処理数を決定できない。

【0010】

本発明は、上記事情に鑑みなされたものであり、その目的は、ジョブを実行するジョブ実行サーバと、ジョブ実行サーバとネットワークを介して接続され、ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤において、ジョブ実行サーバでのジョブの実行に適した処理の並列数を適切に決定することのできる技術を提供することにある。

【課題を解決するための手段】

【0011】

上記目的を達成するため、一観点に係る管理計算機は、ジョブを実行するジョブ実行サーバと、前記ジョブ実行サーバとネットワークを介して接続され、前記ジョブによる処理に使用されるデータを格納するストレージ装置とを備えるデータ処理基盤を管理する管理計算機であって、前記管理計算機は、記憶デバイスと、前記記憶デバイスに接続されたプロセッサとを備え、前記記憶デバイスは、前記データ処理基盤の前記ジョブ実行サーバと前記ストレージ装置との間の通信に関わる構成要素の最大のリソース量の情報である最大リソース量情報と、前記データ処理基盤の前記ストレージ装置のデータへのパスの情報であるパス情報と、前記データ処理基盤の前記構成要素の負荷の情報である負荷情報とを記憶し、前記プロセッサは、前記最大リソース量情報と、前記パス情報と、前記負荷情報とに基づいて、所定のジョブの実行に関わる、前記ジョブ実行サーバから前記ストレージ装置のデータへのパスを構成する構成要素の空きリソース量を計算し、前記空きリソース量に基づいて、前記ジョブ実行サーバにおける前記所定のジョブの実行時における前記ジョブで使用する並列実行可能な処理単位に対する並列して実行可能な数である並列可能数を決定する。

【発明の効果】

【0012】

本発明によれば、データ処理基盤において、ジョブ実行サーバでのジョブの実行に適した処理の並列数を適切に決定することができる。

【図面の簡単な説明】

【0013】

【図1】図1は、第1実施形態に係るデータ分析基盤管理システムの論理的な全体構成図である。

【図2】図2は、第1実施形態に係るデータ分析基盤管理システムの物理的な構成を含む全体構成図である。

10

20

30

40

50

- 【図3】図3は、第1実施形態に係る構成情報記憶部のテーブルの構成図である。
- 【図4】図4は、第1実施形態に係る負荷情報記憶部の負荷情報テーブルの構成図である。
- 【図5】図5は、第1実施形態に係る応答時間情報テーブルの構成図である。
- 【図6】図6は、第1実施形態に係るデータ属性情報テーブルの構成図である。
- 【図7】図7は、第1実施形態に係るプロセス種別情報テーブルの構成図である。
- 【図8】図8は、第1実施形態に係る入力画面の一例を示す図である。
- 【図9】図9は、第1実施形態に係る登録ジョブ情報テーブルの構成図である。
- 【図10】図10は、第1実施形態に係る空きリソース量計算処理のフローチャートである。
- 【図11】図11は、第1実施形態に係る要求リソース量計算処理のフローチャートである。
- 【図12】図12は、第1実施形態に係る最大並列数計算処理のフローチャートである。
- 【図13】図13は、第1実施形態に係る出力画面の一例を示す図である。
- 【図14】図14は、ジョブの要求リソース量の変化を示す図である。
- 【図15】図15は、第2実施形態に係る登録ジョブ情報テーブルの構成図である。
- 【図16】図16は、第2実施形態に係る最大並列数計算処理のフローチャートである。
- 【図17】図17は、第2実施形態に係る最大並列数計算処理の本質を説明する図である。
- 【図18】図18は、第2実施形態に係る完了予定時刻計算処理のフローチャートである。
- 【図19】図19は、第2実施形態に係る出力画面の一例を示す図である。
- 【図20】図20は、第3実施形態に係る入力画面の一例を示す図である。
- 【図21】図21は、第3実施形態に係る登録ジョブ情報テーブルの構成図である。
- 【図22】図22は、第3実施形態に係る最大並列数計算処理のフローチャートである。
- 【発明を実施するための形態】

【0014】

本発明の以下の説明において、開示の一部をなす添付図面を参照するが、これらは本発明を実施できる例示的な実施形態を示すものであって本発明を限定するものではない。これらの図面において、複数の図を通じて同一の符号は同一の構成要素を示している。更に、詳細な説明は各種の例示的な実施形態を提供するが、以下に記述および図示するように、本発明は本明細書に記述および図示する実施形態に限定されるものではなく、当業者には公知または将来公知となる他の実施形態に拡張できる点に注意されたい。

【0015】

また、以下の説明において、本発明を完全に理解されるよう多くの具体的な詳細事項を開示している。しかし、当業者には明らかなように、本発明を実施するためにこれらの具体的な詳細事項のすべてが必要な訳ではない。他の状況において、本発明を無用に分かり難くしないよう、公知の構造、材料、回路、処理およびインタフェースについては詳細に記述せず、および/またはブロック図の形式で示す場合がある。

【0016】

また、以下の説明では、「プログラム」を動作主体として処理を説明する場合があるが、プログラムは、プロセッサ（例えばCPU（Central Processing Unit））によって実行されることで、定められた処理を、適宜に記憶デバイス（例えばメモリ）及び/又はインタフェースデバイス等を用いながら行うため、処理の主体が、プロセッサ（或いは、そのプロセッサを有する装置又はシステム）とされてもよい。また、プロセッサは、処理の一部または全部を行うハードウェア回路を含んでもよい。プログラムは、プログラムソースから計算機のような装置にインストールされてもよい。プログラムソースは、例えば、プログラム配布サーバまたは計算機が読み取り可能な記憶メディアであってもよい。また、以下の説明において、2以上のプログラムが1つのプログラムとして実現されてもよいし、1つのプログラムが2以上のプログラムとして実現されてもよい。

【0017】

また、以下の説明では、計算機、サーバ、コンピュータと記載する場合は、物理的なコ

10

20

30

40

50

ンピュータであってもよいし、仮想化技術などで物理的なコンピュータを仮想的に分割した仮想マシンやコンテナであってもよい。

【0018】

また、以下の説明では、同種の要素を区別して説明する場合は、その要素の参照符号を使用し、同種の要素を区別しないで説明する場合は、その要素の参照符号のうちの共通の親符号を使用することがある。例えば、サーバを特に区別しないで説明する場合には、サーバ150と記載し、個々のサーバを区別して説明する場合には、サーバ150a、150bのように記載することがある。

【0019】

また、以下では、抽出・加工・転送する処理のうち1つまたは複数を組み合わせた処理をETL処理と呼ぶことがある。

【0020】

また、ITシステムを構成する物理的または仮想的な計算機、ネットワーク、ストレージ、OS (Operating System)、ミドルウェア等を総称してITインフラと呼ぶことがある。

【0021】

また、以下の説明では、「AAAテーブル」の表現にて情報を説明することがあるが、情報は、どのようなデータ構造で表現されていてもよい。すなわち、情報がデータ構造に依存しないことを示すために、「AAAテーブル」を「AAA情報」と呼ぶことができる。

【0022】

第1実施形態

図1は、第1実施形態に係るデータ分析基盤管理システムの論理的な全体構成図である。

【0023】

データ分析基盤管理システム1は、管理システムの一例であり、データ分析者が指定したデータを処理するデータ分析基盤100と、データ分析基盤を管理する管理計算機200とを備える。

【0024】

データ分析基盤100は、データ処理基盤の一例であり、データ分析に用いるデータを保存するストレージ装置130と、指定された範囲のデータを転送するための1以上のRDBMS (relational database management system) サーバ120 (120a、120b、120c) と、データに対して所定の処理 (例えば、ETL処理) を実行する1以上のジョブ実行サーバ110 (110d、110e) とを備える。ストレージ装置130と、RDBMSサーバ120と、ジョブ実行サーバ110は、ネットワーク140 (図2参照) を介して接続されている。ジョブ実行サーバ110と、RDBMSサーバ120とは、物理的なコンピュータ (計算機) であってもよいし、仮想化技術などで物理的なコンピュータを仮想的に分割した仮想マシンやコンテナであってもよい。

【0025】

データ分析基盤100においては、ジョブ実行サーバ110と、RDBMSサーバ120とは、パブリッククラウド環境101に配置され、ストレージ装置130はオンプレミス環境102に配置されている。ジョブ実行サーバ110、RDBMSサーバ120、及びストレージ装置130との配置についてはこれに限定せず、いずれのITインフラをオンプレミス環境あるいはパブリッククラウド環境に配置してもよい。

【0026】

ストレージ装置130は、1以上のI/Oポート131 (131a、131b) と、ディスクを仮想的に分割した1以上のボリューム132 (132a、132b、132c) とを有する。I/Oポート131は、ネットワークを介してサーバ (ジョブ実行サーバ110、RDBMSサーバ120) にデータを転送するためのインタフェースである。ボリューム132は、ジョブ実行サーバ110におけるデータ分析処理に利用するデータを格納する記憶装置である。例えば、ボリューム132aには、RDBMSサーバDB1 (RDBMSサーバ120a) に管理されるテーブルDB1__Table1 (DB1__Tab

10

20

30

40

50

1 e 1 3 3 a) が格納される。

【 0 0 2 7 】

R D B M S サーバ 1 2 0 は、ネットワークインタフェースデバイス（ネットワーク I / F ） 1 5 3 を有する。ネットワーク I / F 1 5 3 は、ジョブ実行サーバ 1 1 0 からの要求に基づいてボリューム 1 3 2 からデータを受信、あるいは、ボリューム 1 3 2 へデータを送信するためのインタフェースである。

【 0 0 2 8 】

例えば、データ分析基盤 1 0 0 では、データ分析者が、図示しない計算機等を用いて E T L 処理を実施するデータをストレージ装置 1 3 0 内にあるデータから選択し、処理内容を決定すると、ジョブ実行サーバ 1 1 0 a が R D B M S サーバ 1 2 0 a を介してボリューム 1 3 2 a から指定されたデータを取得し、加工処理をした後、R D B M S サーバ 1 2 0 c を介してボリューム 1 3 2 c に加工後のデータを格納する。ここで、所定のデータに対して実行する E T L 処理をジョブと呼ぶことがある。また、E T L 処理を構成する各々のデータ抽出・加工処理をプロセスと呼ぶことがある。

10

【 0 0 2 9 】

管理計算機 2 0 0 は、構成情報記憶部 3 0 0 と、負荷情報記憶部 4 0 0 と、応答時間情報記憶部 5 0 0 と、データ属性情報記憶部 6 0 0 と、プロセス種別情報記憶部 7 0 0 と、登録ジョブ情報記憶部 8 0 0 とを有する。

【 0 0 3 0 】

構成情報記憶部 3 0 0 は、管理対象のデータ分析基盤 1 0 0 を構成する I T インフラの構成情報を記憶する。負荷情報記憶部 4 0 0 は、データ分析基盤 1 0 0 を構成する I T インフラの構成要素のそれぞれの負荷の時系列データを記憶する。応答時間情報記憶部 5 0 0 は、データ転送のための R D B M S サーバ 1 2 0 からボリューム 1 3 2 の所定のサイズのデータを読み込む、あるいは、書き込むためにかかる時間（応答時間）を記憶する。データ属性情報記憶部 6 0 0 は、データ分析処理に使用するストレージ装置 1 3 0 内に格納されたデータの属性情報を記憶する。

20

【 0 0 3 1 】

構成情報記憶部 3 0 0 、負荷情報記憶部 4 0 0 、応答時間情報記憶部 5 0 0 、及びデータ属性情報記憶部 6 0 0 に格納される情報は、管理対象情報収集プログラム 5 2 0 0 0 を実行する管理計算機 2 0 0 のプロセッサによってデータ分析基盤 1 0 0 から収集され、任意のタイミングで更新される。

30

【 0 0 3 2 】

プロセス種別情報記憶部 7 0 0 は、ジョブ実行サーバ 1 1 0 が実行する処理の内容に応じて分類されたプロセスにおける、所定の処理単位の 1 回の処理の実行に要する処理計算時間と、処理単位で扱うデータ単位に関する情報を記憶する。登録ジョブ情報記憶部 8 0 0 は、データ分析者によって入力部 5 1 1 0 0 から登録されたジョブの情報を記憶する。

【 0 0 3 3 】

次に、本実施形態の管理計算機 2 0 0 の処理概要について説明する。

【 0 0 3 4 】

管理計算機 2 0 0 は、新たなジョブが登録されたことを検知した場合、ジョブ実行の遅延を検知した場合、実行中のジョブのデータ転送パスの負荷が大きく変動したことを検知した場合、あるいは、任意のタイミングにおいて、空きリソース量計算プログラム 9 0 0 を起動して実行する。空きリソース量計算プログラム 9 0 0 （厳密には、空きリソース量計算プログラム 9 0 0 を実行する管理計算機 2 0 0 の C P U 2 1 1 （図 2 参照））は、登録ジョブ情報記憶部 8 0 0 に記憶された所定のジョブ（例えば、新しく登録されたジョブ）の情報を取得し、ジョブ実行時にデータを転送するためのパスの情報（パス情報）と、パス上の構成要素の最大性能（最大リソース量）と、パス上の構成要素それぞれの負荷とを、構成情報記憶部 3 0 0 と負荷情報記憶部 4 0 0 とから取得して、パス上の構成要素の空きリソース量を計算する。

40

【 0 0 3 5 】

50

次に、要求リソース量計算プログラム1000（厳密には、要求リソース量計算プログラム1000を実行するCPU211）は、所定のジョブ情報に関連するパスにおける応答時間と、ジョブが処理するデータのデータ属性情報と、ジョブを構成するプロセスが属するプロセス種別の情報とに基づいて、プロセスでの1処理単位のデータを1回処理する際のデータ転送において、パス上の構成要素にかかる負荷を計算する。次に、最大並列数計算プログラム1100は、空きリソース量計算プログラム900が導出したパス上の各構成要素の空きリソース量と、要求リソース量計算プログラム1000が導出したパス上の各構成要素の1処理単位あたりにかかる各構成要素の負荷とから、プロセスにおいて処理単位の処理を並列で実行可能な最大の数（最大並列数）を計算し、表示部51200に出力する。なお、表示部51200が表示する情報は、ネットワークを介して接続された外部のコンピュータのデバイスに表示するようにしてもよい。

10

【0036】

管理計算機200によると、例えば、新しいジョブが登録されると、データ転送のためのITインフラの負荷から、新しいジョブ実行時にITインフラがボトルネックにならないプロセスの最大並列数が導出される。これにより、データ分析者はジョブ実行時に、ジョブが最も早く完了し、かつ、ジョブ実行サーバ110の処理においてITインフラがボトルネックにならないように、ジョブ実行サーバ110の性能や数量を設定できる。このため、パブリッククラウド環境101におけるジョブ実行サーバ110の使用にかかる課金額を小さくするようにすることができる。なお、ジョブ実行サーバ110の性能や数量については、データ分析者が設定せずに、管理計算機200がデータ分析者によらず自動で設定するようにしてもよい。

20

【0037】

より具体的には、ジョブ実行サーバ2(110e)が、I/Oポート131aを経由して、DBテーブルDB2__Table1のデータを逐次読み込んで処理している場合において、DB1__Table1~DB1__Table999の各テーブルを逐次読み込むジョブ実行サーバ1(110d)が起動された際、ジョブ実行サーバ2(110b)によるI/Oポート131aの負荷を考慮して、I/Oポート131aがボトルネックとならないようにジョブ実行サーバ1(110d)の処理の並列数（並列処理数）を決定できる。

【0038】

データ分析基盤管理システム1のより具体的な構成について説明する。

30

【0039】

図2は、第1実施形態に係るデータ分析基盤管理システムの物理的な構成を含む全体構成図である。

【0040】

データ分析基盤100は、1以上のサーバ150(150a~150e)と、ストレージ装置130とを有する。サーバ150とストレージ装置130とは、ネットワーク140を介して通信可能に接続されている。ネットワーク140は、例えば、アマゾン社のAWS Direct Connect（登録商標）のようにオンプレミス環境とパブリッククラウド環境とを専用線で通信可能にし、かつ、ユーザ毎にネットワーク帯域を仮想的に分割して提供されるネットワークサービスのネットワークであってよい。このネットワークサービスにおいては、ユーザに対して、利用できるネットワークの最大帯域が定義されていてもよい。

40

【0041】

管理計算機200は、例えば、汎用計算機で構成され、プロセッサの一例としてのCPU211と、メモリ212と、ディスク213と、入力デバイス214と、ネットワークインタフェースデバイス（ネットワークI/F）215と、出力デバイス217とを含む。これらのデバイスは、システムバス216を介して接続されている。なお、管理計算機200は、複数のコンピュータにより構成されてもよく、処理効率などに応じて分散及び統合は任意である。

【0042】

CPU211は、メモリ212及び/又はディスク213に格納されているプログラム

50

に従って各種処理を実行する。

【0043】

ディスク213は、記憶デバイスの一例であり、例えば、HDD (Hard Disk Drive) や、SSD (Solid State Drive) 等の不揮発性記憶デバイスである。ディスク213は、構成情報記憶部300と、負荷情報記憶部400と、応答時間情報記憶部500と、データ属性情報記憶部600と、プロセス種別情報記憶部700と、登録ジョブ情報記憶部800とを有する。なお、これらの少なくとも1つの記憶部は、CPU211が参照可能な他の適当な記憶領域に有していてもよい。

【0044】

メモリ212は、記憶デバイスの一例であり、例えば、RAM (RANDOM ACCESS MEMORY) であり、CPU211で実行されるプログラムや、必要な情報を記憶する。メモリ212は、管理対象情報収集プログラム52000と、空きリソース量計算プログラム900と、要求リソース量計算プログラム1000と、最大並列数計算プログラム1100とを記憶する。これらのうちの少なくとも1つのプログラムは、CPU211が参照可能な他の適当な記憶領域に記憶されてよい。また、各プログラムは、コンピュータ読み取り可能な不揮発性の記録媒体に記憶され、読み取り装置によって読み出されたり、ネットワークI/F215を介して外部の装置から取得されたりしてもよい。

【0045】

ネットワークI/F215は、ネットワーク140を介して他の装置(サーバ150、ストレージ装置130)との間で通信を行う。ネットワークI/F215は、例えば、サーバ150及びストレージ装置130等の管理計算機200の管理対象の装置から構成情報、負荷情報等の各種情報を取得する。

【0046】

出力デバイス217は、例えば、ディスプレイ、プリンタ等のデバイスであり、各プログラムが導出する、あるいは、ディスク213に記憶された各種情報を出力(典型的には表示)する。入力デバイス214は、例えば、キーボード、ポインタデバイス等のデバイスであり、ユーザの指示入力を受け付ける。

【0047】

サーバ150は、例えば、汎用計算機であり、メモリ152と、ネットワークI/F153と、CPU151とを含む。サーバ150は、更に、HDDのような不揮発性記憶デバイスで構成されたディスクを有してもよい。CPU151は、メモリ152及び/又は図示しないディスクに格納されているプログラムに従って各種処理を実行する。メモリ152は、例えば、RAMであり、CPU151で実行されるプログラムや、必要な情報を記憶する。ネットワークI/F153は、ネットワーク140を介して他の装置(ストレージ装置130、サーバ150、管理計算機200等)との間で通信を行う。

【0048】

サーバ150には、ジョブ実行サーバ110(110d、110e)を構成するサーバ(150d、150e)と、RDBMSサーバ120を構成するサーバ(150a、150b、150c)とを含む。

【0049】

ジョブ実行サーバ110(110d、110e)を構成するサーバ(150d、150e)のメモリ152は、データ分析者に登録されたジョブを実行するジョブ実行プログラム111と、ジョブの処理の並列実行を制御する並列処理管理プログラム112とを記憶する。ジョブ実行サーバ110(110d、110e)を構成するサーバ(150d、150e)は、例えば、管理計算機200から要求された場合、ネットワーク140を介してサーバ150の構成情報、負荷情報、応答時間情報等を送信する機能を有している。

【0050】

RDBMSサーバ120を構成するサーバ(150a、150b、150c)のメモリ152は、指定されたデータを取得して転送するRDBMSソフトウェア121を記憶する。RDBMSサーバ120を構成するサーバ(150a、150b、150c)は、例

10

20

30

40

50

例えば、管理計算機 200 から要求された場合、ネットワーク 140 を介してサーバ 150 の構成情報、負荷情報、応答時間情報等を送信する機能や、管理するデータのデータ属性情報を送信する機能を有している。

【0051】

ストレージ装置 130 は、サーバ 150 上で動作するプログラム用の記憶領域（論理ボリューム）を提供する装置である。ストレージ装置 130 は、1 以上の I/O ポート 131 と、1 以上のボリューム 132（図では、132a、132b、132c）と、CPU 等のストレージプロセッサ 134 と、を備える。

【0052】

I/O ポート 131 は、ネットワーク 140 を介して接続された装置（例えば、サーバ 150、管理計算機 200 等）と通信するためのインタフェースである。

10

【0053】

ボリューム 132 は、データ分析に利用するデータを格納する記憶装置である。ボリューム 132 は、ディスクを仮想的に分割した記憶装置である。ボリューム 132 を構成するディスクは、1 以上の HDD、SSD 等の不揮発性記憶デバイスであってもよい。ボリューム 132 は、複数の HDD で構成された RAID（Redundant Array of Independent (or Inexpensive) Disks）グループであってもよい。例えば、ボリューム 132a には、RDBMS サーバ DB1 に管理されるテーブル DB1__Table1 が格納される。ボリューム 132 上のデータはボリューム 132 に割り当てられた 1 以上の I/O ポート 131 を介して転送される。

20

【0054】

なお、ストレージ装置 130 は、サーバ 150 に対して記憶領域としてボリュームを提供してもよい。この場合には、ストレージ装置 130 は、ストレージ装置 130 の構成情報、負荷情報等を管理計算機 200 に送信してもよい。

【0055】

< 構成情報記憶部 300 >

図 3 は、第 1 実施形態に係る構成情報記憶部のテーブルの構成図である。

【0056】

構成情報記憶部 300 は、構成要素情報テーブル 310 と、パス情報テーブル 320 とを格納する。

30

【0057】

構成要素情報テーブル 310 は、データ分析基盤 100 を構成する IT インフラの構成要素の最大のリソース量（性能値）を記憶するテーブルであり、構成要素ごとのエントリを格納する。

【0058】

構成要素情報テーブル 310 のエントリは、構成要素 ID 311、監視メトリック種別 312、及び最大性能値 313 とのフィールドを備える。構成要素 ID 311 には、管理対象のデータ分析基盤 100 を構成する IT インフラの構成要素を一意に識別する値（構成要素 ID）が格納される。監視メトリック種別 312 には、構成要素の性能の監視される項目を識別する値（メトリック種別、監視メトリック種別）が格納される。最大性能値 313 には、エントリに対応する構成要素 ID の構成要素が、エントリに対応するメトリック種別の項目の性能についての最大値（最大性能値：最大リソース量）が格納される。最大性能値 313 には、最大性能値の大きさを示す単位情報が含まれていてもよい。最大性能値 313 に格納される最大性能値は、構成要素の物理的な限界値が格納されていてもよいし、性能障害が発生しないように物理的な限界値に対してマージンを取った値でもよい。

40

【0059】

例えば、エントリ 3101 は、以下の内容を示す。すなわち、構成要素 ID が「I/O ポート 131a」の構成要素（ここでは、I/O ポート 131a）のデータ受信時の転送速度（受信転送速度）の最大性能値は 10 Gbps であることを示す。

【0060】

50

パス情報テーブル320は、所定のデータをジョブ実行サーバ110に転送する、または、ジョブ実行サーバ110から転送するためにデータが入出力されるパスにおける構成要素のリスト（パス情報）を記憶するテーブルであり、データを格納するデータ群ごとのエントリを記憶する。パス情報テーブル320のエントリは、データID321、ネットワークI/FID322、ネットワークID323、I/OポートID324、及びボリュームID325のフィールドを備える。

【0061】

データID321には、管理対象のデータ分析基盤100においてデータ分析者が利用できるように分類、分割されたデータ群、及び、データの格納先を一意に識別する値（データID）が格納される。例えば、上場企業ごとに作成され、1行（1エントリ）ごとに日付と、その1日の始値と終値とが記憶された株価の分析のための1つのデータベーステーブルを1つのデータ群としてもよい。

10

【0062】

ネットワークI/FID322には、エントリに対応するデータIDが示すデータ群のデータをジョブ実行サーバ110に転送するサーバ（例えば、RDBMSサーバ120）のネットワークI/Fを一意に識別する値（ネットワークI/FID）が格納される。ネットワークID323には、エントリに対応するデータIDが示すデータ群のデータをジョブ実行サーバ110に転送するネットワークを一意に識別する値（ネットワークID）が格納される。I/OポートID324には、エントリに対応するデータID321が示すデータ群のデータをジョブ実行サーバ110に転送するストレージ装置130のI/Oポート131を一意に識別する値（I/OポートID）が格納される。ボリュームID325には、エントリに対応するデータID321が示すデータ群のデータが格納されたボリューム132を一意に識別する値（ボリュームID）が格納される。

20

【0063】

例えば、パス情報テーブル320のエントリ3201は、データIDが「DB1__Table1」のデータ群のデータをジョブ実行サーバ110に転送するパスは、「ネットワークI/F121a」と、「ネットワーク140」のネットワークと、「I/Oポート131a」とを經由して「ボリューム132a」から転送されるパスであることを示す。なお、1つのデータIDのデータ群に対して、複数のパスがあってもよい。例えば、1つのボリューム132に対して複数のI/Oポート131が割り当てられ、いずれかのI/Oポート131を介してデータが転送される場合には、パスは複数になる。また、パス情報テーブル320は、エントリ3201のようにETL処理に使用するデータを読み出すパスだけでなく、エントリ3204のようにETL処理後のデータを格納するためのパスも記憶される。また、本実施形態においては、パス情報はデータ、ネットワークI/F、ネットワーク、I/Oポート、ボリュームで構成したが、ボトルネックとなり得るそのほかの構成要素を入れてもよい。例えば、データを転送する際に負荷が発生するストレージプロセッサやRDBMSサーバのCPUなどを含めてもよい。

30

【0064】

なお、本実施形態においては、構成要素情報テーブル310やパス情報テーブル320に格納される構成要素IDは、物理的な構成要素を識別する値であっても、物理的な構成要素を仮想的に統合、あるいは、分割した構成要素を識別する値であってもよい。例えば、複数のI/Oポートが仮想的に統合されている場合は、統合されている仮想的なI/Oポートの1つの識別子としてもよい。逆に、1つのI/Oポートを複数の仮想的なI/Oポートに分割し、それぞれに対してリソースの上限値を設定している場合には、それぞれの仮想的なI/Oポートのそれぞれの識別子としてもよい。また、RDBMSソフトウェアなどによってRDBMSサーバ120やボリューム132が仮想化されている場合も同様に複数の物理的な構成要素を1つの構成要素とした識別子としてもよい。

40

【0065】

<負荷情報記憶部400>

負荷情報記憶部400は、管理対象情報収集プログラム52000によって収集された

50

データ分析基盤 100 の各構成要素における、監視メトリック種別毎に監視された負荷の時系列推移を記憶する負荷情報テーブルを格納する。

【0066】

図4は、第1実施形態に係る負荷情報記憶部の負荷情報テーブルの構成図である。

【0067】

負荷情報記憶部400は、構成要素と監視メトリック種別との組合せ毎に負荷情報テーブル(410, 420, 430等)を記憶する。なお、本実施形態では、負荷情報記憶部400は、例えば、構成要素情報テーブル310のエントリの数(構成要素と監視メトリック種別との組合せの数)と同じ数の負荷情報テーブルを記憶する。負荷情報テーブル410は、I/Oポート131aの受信転送速度に対応する負荷情報テーブルの一例であり、負荷情報テーブル420は、I/Oポート131aの送信転送速度に対応する負荷情報テーブルの一例であり、負荷情報テーブル430は、ボリューム132aの転送速度に対応する負荷情報テーブルの一例である。

10

【0068】

負荷情報テーブル(410, 420, 430)のエントリは、時刻401(401a, 401c, 401e)と、観測値402(402b, 402d, 402f)とのフィールドを含む。時刻401には、構成要素の監視メトリック種別における負荷を観測した時刻が格納される。観測値402には、観測した負荷の値が格納される。例えば、負荷情報テーブル410のエントリ4101は、以下の内容を示す。すなわち、I/Oポート131aの受信転送速度の負荷は、2020年1月1日0時0分において2.0Gbpsであったことを示す。

20

【0069】

< 応答時間情報記憶部500 >

応答時間情報記憶部500は、データ転送のためのRDBMSサーバ120からボリューム132に記憶された所定のデータを所定のサイズで読み込む、あるいは、書き込む際にかかる時間(応答時間と呼ぶことがある)の情報を記憶する応答時間情報テーブル510を格納する。例えば、RDBMSサーバ120が、所定のデータを読み書きし、読み書きにおける応答時間を計測する応答時間計測プログラムを実行し、その応答時間計測プログラムにより計測された応答時間の情報を管理対象情報収集プログラム52000が応答時間情報テーブル510に記憶するようにしてもよい。応答時間計測プログラムは、ジョブ実行サーバ110や、ジョブ実行サーバ110と同じ環境にあるその他の計算機で実行されてもよい。応答時間計測プログラムによる応答時間の計測は定期的を実施されてよいし、後述する要求リソース量計算処理(図11参照)を実行する直前に実施されてもよい。

30

【0070】

図5は、第1実施形態に係る応答時間情報テーブルの構成図である。

【0071】

応答時間情報テーブル510は、時刻511と、データID512と、処理種別513と、応答時間514とのフィールドを備える。時刻511には、応答時間を計測した時刻が格納される。データID512には、応答時間を計測のために読み込んだデータ、あるいは、書き込んだデータの格納先を一意に識別する値(格納先ID)が格納される。格納先IDとしては、例えば、読込んだデータのデータ群がデータ格納先に対応する場合には、データ群のデータIDであってもよい。処理種別513には、応答時間の計測時に実行した処理の種別(処理種別)、例えば、データの読み込み処理であるか、書き込み処理であるかを識別する値が格納される。応答時間514には、計測した応答時間が格納される。応答時間は、1回の計測における応答時間でもよいし、複数回実施して得られたそれぞれの時間の平均値や最大値でもよい。

40

【0072】

例えば、エントリ5003は、以下の内容を示す。2020年1月1日0時0分にデータIDが「DB3_Table1」のデータ群のデータが、格納先へ所定のサイズのデータを書き込む際にかかった時間が10ms(ミリ秒)であったことを示す。なお、本実施

50

形態においては、応答時間の計測は、データIDのデータ群のデータ、及び、データ格納先ごとに実施しているが、ボリュームごとやRDBMSサーバごとに実施してもよい。また、パスが同じデータに関しては、応答時間の計測を1回にまとめて実施してもよい。

【0073】

<データ属性情報記憶部600>

データ属性情報記憶部600は、管理対象のデータ分析基盤100においてデータ分析者が利用できるように分類、分割されたデータ群の属性（例えば、データの容量）に関する情報（データ属性情報）を記憶するデータ属性情報テーブル610を格納する。

【0074】

図6は、第1実施形態に係るデータ属性情報テーブルの構成図である。

10

【0075】

データ属性情報テーブル610は、データ群ごとのエントリを格納する。データ属性情報テーブル610のエントリは、データID611と、データ容量612とのフィールドを備える。データID611には、データ分析者が利用できるように分類、分割されたデータ群を一意に識別する値（データID）が格納される。データ容量612には、エントリに対応するデータIDのデータ群の容量（データ容量）が格納される。

【0076】

例えば、エントリ6001は、以下の内容を示す。すなわち、データIDが「DB1__Table1」のデータ群のデータのデータ容量が50MBであることを示す。

【0077】

20

本実施形態においては、以降で、1つのRDBテーブルを並列可能な処理単位の1単位としてETL処理を実行する例を挙げるため、データ属性情報テーブル610では、1つのRDBテーブルを単位としてデータ容量を記憶している。ETL処理を実行するデータの単位がRDBテーブルの行単位の可能性がある場合には、データ属性情報テーブル610に、行単位のデータ容量、あるいは、その行単位のデータの平均値も併せて格納してもよい。また、本実施形態では、RDBのデータを対象としているので、RDBテーブルのデータ容量を記憶するようにしていたが、例えば、ETL処理を実行する対象のデータがファイルである場合は、1ファイルごとのデータ容量を記憶するようにしてもよい。また、データがオブジェクトストレージに格納されている場合は、オブジェクトのデータ容量を記憶するようにしてもよい。また、ETL処理の並列処理を効率化するために、RDB

30

テーブル、ファイル、オブジェクトがETL処理ソフトウェアによって論理的なパーティションで分割されて管理されている場合は、パーティション毎のデータ容量、あるいはその平均値を併せて記憶するようにしてもよい。

【0078】

<プロセス種別情報記憶部700>

プロセス種別情報記憶部700は、管理対象のデータ分析基盤100において、ジョブ実行サーバ110が実行し得るジョブのプロセスの内容を分類し、プロセスにおける並列可能な処理単位の1回の実行に対する計算処理時間と、プロセスでの処理単位のデータに関する情報と、を格納するプロセス種別情報テーブル710を記憶する。

【0079】

40

図7は、第1実施形態に係るプロセス種別情報テーブルの構成図である。

【0080】

プロセス種別情報テーブル710は、プロセスの種別毎のエントリを格納する。プロセス種別情報テーブル710のエントリは、プロセス種別ID711と、1回の処理計算時間712と、読込単位713と、書込単位714と、読込回数715と、書込回数716とのフィールドを格納する。

【0081】

プロセス種別ID711には、ジョブ実行サーバ110が実行し得るジョブを構成するプロセスの内容を分類し、一意に識別する値（プロセス種別ID）が格納される。1回の処理計算時間712には、プロセスの実行時に並列して実行可能な処理単位の1回の実行

50

にかかる計算時間（処理計算時間）が格納される。本実施形態においては、この処理計算時間には、処理に必要なデータの転送にかかる時間は含まないものとする。1回の処理計算時間は、データ分析基盤100の構築時などに、あらかじめ試験的にプロセスを実行し、処理計算時間を計測することにより導出された値を用いてもよく、別のデータ分析基盤で同様なプロセスについて計測された値を用いてもよい。また、処理計算時間は、エントリに対応するジョブのプロセスが実行される際に計測し、その都度更新するようによい。

【0082】

読込単位713には、並列実行される処理単位の1回あたりにボリューム132から読み込まれるデータの単位が格納される。書込単位714には、並列実行される処理単位の1回あたりにボリューム132に書き込まれるデータの単位が格納される。読込回数715には、並列実行される処理単位の1回あたりにボリューム132からデータが読み込まれる回数が格納される。書込回数716には、並列実行される処理単位の1回あたりにボリューム132に書き込まれるデータの回数が格納される。

10

【0083】

例えば、エントリ7001は、以下の内容を示す。すなわち、プロセス種別IDが「Table__to__Table__Extract__Column」のプロセスは、並列実行される処理単位の1回あたりにかかる処理計算時間は30ミリ秒であり、並列実行される処理単位の1回あたりに1つのテーブルをボリューム132から読み込み、1つのテーブルをボリューム132に書き込むことを示す。

20

【0084】

なお、本実施形態では、所定のプロセスの1つの処理単位あたりの処理計算時間を一意の値として用いているが、処理内容によっては一意に決定しない場合がある。例えば、1回の処理単位の処理で扱うデータ容量やジョブ実行サーバ110の性能によって大きく変動する場合がある。この場合、プロセス種別ごとにデータ容量と1つの処理単位あたりの処理計算時間とのモデル（例えば、データ容量と1処理単位あたりの処理計算時間の相関関係を表した計算式）や、ジョブ実行サーバと1処理単位あたりの処理計算時間とのモデルを用意し、データ属性情報テーブル610の情報に基づいて、1処理単位あたりの処理計算時間を、後述する要求リソース量計算処理（図11参照）を実行するタイミング、あるいは、ジョブが登録されるタイミングで計算するようによい。また、本実施形態においては、処理単位で処理するデータに関する情報をテーブルに記憶するようにしているが、本発明はこれに限られず、例えば、後述する空きリソース量計算処理（図10参照）等の処理を実行する際に、実行する処理プログラムのソースコードを解析して同等の情報を抽出して利用するようによい。

30

【0085】

<入力画面>

図8は、第1実施形態に係る入力画面の一例を示す図である。図8に示す入力画面は、GUI（Graphical User Interface）で実装した場合の一例を示す。

【0086】

入力画面51110は、入力部51100によりデータ分析者が、ETL処理を実行する入力データと、実行するETL処理内容と、処理後のデータの格納先と、その関係を入力するための入力画面である。入力画面51110は、入力領域51111を有する。入力領域51111は、データノード51112と、プロセスノード51113と、アウトプットノード51114とを有する。データノード51112は、ジョブに対してETL処理を実行するデータソースを定義する領域である。プロセスノード51113は、プロセスにおいて実行する処理を定義する領域である。アウトプットノード51114は、処理したデータの保存先を定義する領域である。

40

【0087】

入力画面51110は、ジョブAが、RDBMSサーバ120a（DB1）が管理するIDがDB1__Table1～999である999個のテーブルを入力として、テーブル

50

のカラムのうち、「Date」、「Opening Price」、「Closing Price」を抽出し、RDBMSサーバ120c(DB3)が管理するDB3_Table1のテーブルに格納することを示す。

【0088】

<登録ジョブ情報記憶部800>

登録ジョブ情報記憶部800は、入力部51100によって入力された情報に基づく、実行予定、あるいは実行中のジョブによって実行するプロセスと、データソースと、アウトプットとの情報を格納する登録ジョブ情報テーブル810を記憶する。

【0089】

図9は、第1実施形態に係る登録ジョブ情報テーブルの構成図である。

10

【0090】

登録ジョブ情報テーブル810は、ジョブごとのエントリを格納する。登録ジョブ情報テーブル810のエントリは、ジョブID811と、プロセスID812と、プロセス種別ID813と、パラメータ814と、データソース815と、アウトプット816とのフィールドを格納する。

【0091】

ジョブID811には、データ分析者が登録したジョブ(登録ジョブ)を一意に識別する値(ジョブID)が格納される。プロセスID812には、ジョブが実行するプロセス(処理)を一意に識別する値(プロセスID)が格納される。プロセス種別ID813には、プロセスの種別を一意に識別する値(プロセス種別ID)が格納される。パラメータ814は、プロセスにおける設定値(パラメータ)が格納される。パラメータ814には、例えば、入力画面51110のプロセスノード51113で設定された設定値が格納される。データソース815は、プロセスに入力されるデータを識別する値が格納される。データソース815には、例えば、入力画面51110のデータノード51112で設定した値が格納される。アウトプット816には、プロセスで出力されるデータの保存先を一意に識別する値が格納される。アウトプット816には、例えば、入力画面51110のアウトプットノード51114で設定した値が格納される。

20

【0092】

例えば、エントリ8001は、以下の内容を示す。すなわち、ジョブIDが「ジョブA」のジョブにおいて、プロセス種別が「Table__to__Table__Extract__Column」の「プロセスa」は設定値「Date, Opening Price, Closing Price」が設定されて実行され、プロセスaへの入力データは「DB1_Table1~DB1_Table999」の999個のテーブルであり、処理後のデータの保存先は「DB3_Table1」のテーブルであることを示す。

30

【0093】

なお、本実施形態においては、1つのジョブに対して入力データと、プロセスと、アウトプットとが1つずつ定義されているが、1つのジョブに対して複数定義されていてもよい。例えば、ジョブにおいて、プロセスaを実行した後に処理結果を入力としたプロセスbが定義され、プロセスbの結果がストレージ装置130内のボリューム132に保存される場合には、複数のプロセスと、それぞれのプロセスの入力データとアウトプットとをエントリに格納するようにしてもよい。

40

【0094】

次に、データ分析基盤管理システム1における処理動作について詳細に説明する。

【0095】

<空きリソース量計算処理>

空きリソース量計算処理は、管理計算機200の空きリソース量計算プログラム900をCPU211が実行することにより行われる処理であり、ジョブが処理するデータのデータ転送パスを構成する構成要素それぞれの空きリソース量を計算する処理である。

【0096】

図10は、第1実施形態に係る空きリソース量計算処理のフローチャートである。

50

【 0 0 9 7 】

空きリソース量計算処理は、例えば、新たなジョブが登録されたことを検知した場合、ジョブ実行の遅延を検知した場合、実行中のジョブのデータ転送パスの負荷が大きく変動したことを検知した場合、あるいは、任意のタイミングにおいて、処理が開始される。なお、空きリソース量計算処理は、後述の要求リソース量計算処理と順序を入れ替えて実行されてもよく、あるいは、同時に実行されてもよい。

【 0 0 9 8 】

ステップ S 9 0 1 において、空きリソース量計算プログラム 9 0 0 (厳密には、空きリソース量計算プログラム 9 0 0 を実行する CPU 2 1 1) は、登録ジョブ情報テーブル 8 1 0 から所定のジョブに関連するエントリを取得する。例えば、「所定のジョブ」は、空きリソース量計算処理を開始するトリガとなった新しく登録されたジョブ (登録ジョブ) 、処理の遅延が検知されたジョブ、データ転送パスの負荷が大きく変動したジョブの 1 つ、あるいは、任意のジョブであってよい。

10

【 0 0 9 9 】

ステップ S 9 0 2 において、空きリソース量計算プログラム 9 0 0 は、ステップ S 9 0 1 で取得したエントリのデータソース 8 1 5 、及びアウトプット 8 1 6 に格納された識別子に関連するパス情報、すなわち、識別子のデータ (例えば、テーブル) にアクセスするためのパス情報を構成情報記憶部 3 0 0 のパス情報テーブル 3 2 0 から取得する。

【 0 1 0 0 】

ステップ S 9 0 2 の後に、空きリソース量計算プログラム 9 0 0 は、ステップ S 9 0 2 で取得したパス情報のそれぞれを処理対象として、ループ 1 の処理 (ステップ S 9 0 3 ~ S 9 0 6) を繰り返す。ここで、処理対象のパス情報を対象パス情報という。

20

【 0 1 0 1 】

ステップ S 9 0 3 において、空きリソース量計算プログラム 9 0 0 は、対象パス情報のエントリが示す構成要素の ID (ネットワーク I / F ID、ネットワーク ID、I / O ポート ID、ボリューム ID) に基づいて、構成情報記憶部 3 0 0 の構成要素情報テーブル 3 1 0 からそれぞれの構成要素に関連するエントリをすべて取得する。なお、構成要素に関連するエントリには、構成要素の最大性能値の情報が含まれている。

【 0 1 0 2 】

ステップ S 9 0 4 において、空きリソース量計算プログラム 9 0 0 は、対象パス情報のエントリが示す構成要素の ID (ネットワーク I / F ID、ネットワーク ID、I / O ポート ID、ボリューム ID) に基づいて、負荷情報記憶部 4 0 0 の構成要素に関連する負荷情報テーブル参照し、負荷情報テーブルから負荷情報のエントリを取得し、各構成要素の各監視メトリック種別の負荷を導出する。この時、取得するエントリは、時刻 4 0 1 が最新のエントリであって、かつ負荷の値としてもよい。または、例えば、時刻 4 0 1 が所定の期間に含まれるエントリすべてを取得し、それらの負荷の値の平均値、最大値、あるいは、平均値に標準偏差を足した値を負荷の値としてもよい。また、公知の負荷予測アルゴリズムなどを用いて、所定の期間の負荷から将来の負荷を予測し、予測した負荷の値を、各構成要素の負荷の値としてもよい。

30

【 0 1 0 3 】

ステップ S 9 0 5 において、空きリソース量計算プログラム 9 0 0 は、ステップ S 9 0 3 で取得した構成要素情報テーブル 3 1 0 のエントリすべてに対し、最大性能値 3 1 3 の最大性能値から、ステップ S 9 0 4 で導出した対応する負荷の値を減算することにより、各構成要素の各監視メトリック種別についての空きリソース量を計算する。

40

【 0 1 0 4 】

ステップ S 9 0 6 において、空きリソース量計算プログラム 9 0 0 は、ステップ S 9 0 5 で計算した、構成要素と監視メトリック種別とのすべての組に対する空きリソース量を記憶する。例えば、空きリソース量を、メモリ 2 1 2 に記憶してよい。

【 0 1 0 5 】

空きリソース量計算処理の具体例は以下のとおりである。例えば、図 8 の入力画面 5 1

50

110で示すように新規にジョブAが登録された場合には、ステップS901において、空きリソース量計算プログラム900は、登録ジョブ情報テーブル810のエントリ8001を取得する。次に、空きリソース量計算プログラム900は、エントリ8001のデータソース815の値「DB1__Table1~DB1__Table999」と、アウトプット816の値「DB3__Table1」とに基づいて、パス情報テーブル320の999個のエントリ3201~エントリ3202と、1個のエントリ3204を取得する(ステップS902)。

【0106】

次いで、空きリソース量計算プログラム900は、取得した1000個のエントリの夫々を処理対象としてループ1の処理(ステップS903~S906)を実行する。例えば、処理対象としてエントリ3201を選択した場合、ステップS903では、空きリソース量計算プログラム900は、エントリ3201に格納された構成要素ID「ネットワークI/F121a」、「ネットワーク140」、「I/Oポート131a」、「ボリューム132a」のそれぞれをキーとして構成要素情報テーブル310からエントリを取得する。例えば、「I/Oポート131a」をキーとした場合は、空きリソース量計算プログラム900は、エントリ3101とエントリ3102とを取得する。

【0107】

ステップS904では、空きリソース量計算プログラム900は、エントリ3201に格納された構成要素IDをキーとして、負荷情報記憶部400の対応する負荷情報テーブルからエントリを取得する。例えば、「I/Oポート131a」をキーとし、最新の負荷を構成要素に対する負荷の値とする場合は、空きリソース量計算プログラム900は、負荷情報テーブル410のエントリ4104と、負荷情報テーブル420のエントリ4204と、を取得する。そして、ステップS905では、空きリソース量計算プログラム900は、例えば、I/Oポート131aの受信転送速度の空きリソース量を計算する場合は、エントリ3101の最大性能値である「10Gbps」からエントリ4104の「2.0Gbps」を減算して得られた8.0Gbpsを空きリソース量とし、ステップS906では、I/Oポート131aの受信転送速度と、その空きリソース量との組を記憶する。

【0108】

なお、本実施形態では、各々の処理において、説明のために不要なデータを取得、あるいは、計算している場合がある。例えば、ジョブ実行サーバ110がボリュームからデータを読み込む処理に対して空きリソース量を計算する場合は、I/Oポート131の送信転送速度の空きリソース量のみを計算すればよいが、本実施形態においては受信転送速度の空きリソース量も含めて計算している。このような不要なデータ取得、あるいは、計算に関しては必要に応じて削減してもよい。

【0109】

<要求リソース量計算処理>

要求リソース量計算処理は、管理計算機200の要求リソース量計算プログラム1000をCPU211が実行することにより行われる処理であり、登録されたジョブが実行するプロセスが並列可能な1処理単位(1並列処理単位)あたりに各構成要素に要求するリソース量(要求リソース量)を計算する処理である。

【0110】

図11は、第1実施形態に係る要求リソース量計算処理のフローチャートである。

【0111】

要求リソース量計算処理は、空きリソース量計算処理の完了を検知して開始されてもよい。また、要求リソース量計算処理は、空きリソース量計算処理と順序を入れ替えて実行されてもよく、あるいは、同時に実行されてもよい。この場合、例えば、要求リソース量計算処理は、新たなジョブが登録されたことを検知した場合、ジョブ実行の遅延を検知した場合、実行中のジョブのデータ転送パスの負荷が大きく変動したことを検知した場合、あるいは、任意のタイミングにおいて、開始されてもよい。

【0112】

10

20

30

40

50

ステップ S 1 0 0 1 において、要求リソース量計算プログラム 1 0 0 0（厳密には、要求リソース量計算プログラム 1 0 0 0 を実行する CPU 2 1 1）は、登録ジョブ情報テーブル 8 1 0 から所定のジョブに関連するエントリ（ジョブ情報）を取得する。例えば、所定のジョブとしては、空きリソース量計算処理を開始するトリガとなったジョブであってよい。

【 0 1 1 3 】

ステップ S 1 0 0 1 の後に、要求リソース量計算プログラム 1 0 0 0 は、ステップ S 1 0 0 1 で取得したジョブ情報が有するすべてのプロセスの情報（プロセス情報）を処理対象として、ループ 2 の処理（ステップ S 1 0 0 2 ~ S 1 0 0 6）を繰り返す。ここで、処理対象のプロセス情報を対象プロセス情報という。

10

【 0 1 1 4 】

ステップ S 1 0 0 2 において、要求リソース量計算プログラム 1 0 0 0 は、対象プロセス情報のデータソース 8 1 5 及びアウトプット 8 1 6 に格納された識別子に関連するパス情報、すなわち、識別子のデータにアクセスするためのパス情報を構成情報記憶部 3 0 0 のパス情報テーブル 3 2 0 から取得する。

【 0 1 1 5 】

ステップ S 1 0 0 3 において、要求リソース量計算プログラム 1 0 0 0 は、対象プロセス情報のデータソース 8 1 5、あるいは、アウトプット 8 1 6 に対するデータについての応答時間を応答時間情報テーブル 5 1 0 から取得する。取得する応答時間は、例えば、応答時間情報テーブル 5 1 0 における、データソース 8 1 5、あるいは、アウトプット 8 1 6 の識別子に関連するデータについての最新の応答時間であってよい。なお、要求リソース量計算処理が開始されてからデータソース 8 1 5、あるいは、アウトプット 8 1 6 の識別子に関連するデータについての応答時間を計測することにより取得してもよい。

20

【 0 1 1 6 】

ステップ S 1 0 0 4 において、要求リソース量計算プログラム 1 0 0 0 は、対象プロセス情報のプロセス種別 ID 8 1 3 のプロセス種別 ID に基づいて、プロセス種別情報テーブル 7 1 0 から関連するエントリ（1 回の処理計算時間、読込単位、書込単位、読込回数、書込回数等）を取得する。

【 0 1 1 7 】

ステップ S 1 0 0 5 において、要求リソース量計算プログラム 1 0 0 0 は、対象プロセス情報のデータソース 8 1 5 の値に関連するデータのデータ属性、すなわち、データ容量をデータ属性情報テーブル 6 1 0 から取得する。

30

【 0 1 1 8 】

ステップ S 1 0 0 6 において、要求リソース量計算プログラム 1 0 0 0 は、ステップ S 1 0 0 3 で取得した応答時間と、ステップ S 1 0 0 4 で取得したエントリの情報（1 回の処理計算時間、読込単位、書込単位、読込回数、及び書込回数）と、ステップ S 1 0 0 5 で取得したデータ容量とに基づいて、プロセスの 1 並列処理単位あたりに対する負荷を、データ転送の向きを考慮して、ステップ S 1 0 0 2 で取得したパス情報の各構成要素に対して計算する。例えば、I/Oポート 1 3 1 a の負荷（帯域）を計算する場合、以下の式（1）、（2）で計算してもよい。

40

【 0 1 1 9 】

1 並列処理単位あたりの I/Oポート 1 3 1 a の送信転送速度の負荷（Gbps）

$$= (\text{読込データ容量の平均値} \times \text{読込回数}) / (\text{読込データの応答時間} \times \text{読込回数} + 1 \text{ 回の処理計算時間} + \text{書込データの応答時間} \times \text{書込回数}) \cdots (1)$$

1 並列処理単位あたりの I/Oポート 1 3 1 a の受信転送速度の負荷（Gbps）

$$= (\text{書込データ容量の平均値} \times \text{書込回数}) / (\text{読込データの応答時間} \times \text{読込回数} + 1 \text{ 回の処理計算時間} + \text{書込データの応答時間} \times \text{書込回数}) \cdots (2)$$

【 0 1 2 0 】

50

ここで、読込データ容量の平均値は、データソース 8 1 5 と読込単位 7 1 3 とに基づいて算出される、1 並列処理単位あたりの読込データの平均値でよい。また、書込データ容量の平均値は、例えば、読込データ容量の平均値から計算して求めてもよく、例えば、以下の式 (3) で算出してもよい。

書込データ容量の平均値 = 読込データ容量平均値 × 読込回数 / 書込回数 …… (3)
【 0 1 2 1 】

また、テーブルからファイルに変換する場合は、テーブルにおけるデータ容量に対して一般的なテーブルからファイルへ変換する際の圧縮率をかけてもよい。読み込んだデータの一部を抽出して書き込む場合は、その削減率 (予測値含む) をかけてもよい。

【 0 1 2 2 】

なお、上述の式 (1) , (2) は、データの読み込みや書込みのためのデータ転送パスが同じであることを前提としたものとしているが、例えば、2 つのデータ転送パスからデータを読み込み、かつ、2 つの I / O ポートを経由するデータ数の割合が 3 : 2 である場合は、上述の式 (1) で計算した「1 並列処理単位あたりのネットワーク 1 4 0 の上り転送速度への負荷」に対してそれぞれ 3 / 5 と 2 / 5 とを乗算した値をかけて計算してよい。また、読み込みや書込みのためのデータ転送のパスが異なる場合は、パスごとに読込データ容量の平均値や書込データ容量の平均値を計算して用いてもよい。

【 0 1 2 3 】

ステップ S 1 0 0 7 において、要求リソース量計算プログラム 1 0 0 0 は、ステップ S 1 0 0 1 で取得したジョブの 1 並列処理単位あたりの各構成要素に対する要求リソース量を計算し、メモリ 2 1 2 等に記憶する。例えば、ジョブのプロセスが 1 つである場合は、ステップ S 1 0 0 6 で計算したプロセスの 1 並列処理単位あたりの各構成要素に対する負荷を要求リソース量とすればよい。一方、例えば、ジョブに並列で実行されるプロセスが複数含まれている場合は、ステップ S 1 0 0 6 で計算した複数のプロセスによる負荷を足して要求リソース量とすればよく、また、例えば、ジョブに順に実行される複数のプロセスがある場合は、ステップ S 1 0 0 6 で計算した複数のプロセスによる負荷の最大値を要求リソース量とすればよく、ジョブに並列で実行される複数のプロセスと、順に実行される複数のプロセスとが含まれる場合には、それらを組み合わせて要求リソース量を計算すればよい。

【 0 1 2 4 】

要求リソース量計算処理の具体例は以下のとおりである。例えば、図 8 の入力画面 5 1 1 1 0 で示すように新規にジョブ A が登録された場合には、要求リソース量計算プログラム 1 0 0 0 は、ステップ S 1 0 0 1 において、登録ジョブ情報テーブル 8 1 0 のエン트리 8 0 0 1 を取得する。次いで、要求リソース量計算プログラム 1 0 0 0 は、取得したエントリの数だけループ 2 の処理 (ステップ S 1 0 0 2 ~ S 1 0 0 6) を繰り返す。

【 0 1 2 5 】

ステップ S 1 0 0 3 において、要求リソース量計算プログラム 1 0 0 0 は、エン트리 8 0 0 1 のデータソース 8 1 5 の値「DB 1 __ T a b l e 1 ~ DB 1 __ T a b l e 9 9 9」と、アウトプット 8 1 6 の値「DB 3 __ T a b l e 1」とに基づいて、パス情報テーブル 3 2 0 の 9 9 9 個のエン트리 3 2 0 1 ~ エン트리 3 2 0 2 と、1 個のエン트리 3 2 0 4 とを取得する (ステップ S 1 0 0 3) 。

【 0 1 2 6 】

次に、要求リソース量計算プログラム 1 0 0 0 は、登録ジョブ情報テーブル 8 1 0 のエン트리 8 0 0 1 のデータソース 8 1 5 の値「DB 1 __ T a b l e 1 ~ DB 1 __ T a b l e 9 9 9」と、アウトプット 8 1 6 の値「DB 3 __ T a b l e 1」とに基づいて、応答時間情報テーブル 5 1 0 からエン트리 5 0 0 1 とエン트리 5 0 0 3 とを参照し、DB 1 の読み込み応答時間 3 m s (3 ミリ秒) と、DB 3 __ T a b l e 1 への書き込み応答時間 1 0 m s とを取得する (ステップ S 1 0 0 3) 。

【 0 1 2 7 】

ステップ S 1 0 0 4 では、要求リソース量計算プログラム 1 0 0 0 は、エン트리 8 0 0

10

20

30

40

50

1のプロセス種別「Table__to__Table__Extract__Column」に基づいて、プロセス種別テーブル710からエントリ7001を取得する。ステップS1005において、要求リソース量計算プログラム1000は、エントリ8001のデータソース815の値「DB1__Table1~DB1__Table999」に対応するデータのデータ容量をデータ属性情報テーブル610から取得し、その平均値（この例では、例えば、70MB）を計算する。

【0128】

ステップS1006において、要求リソース量計算プログラム1000は、ステップS1002で取得したパス情報（エントリ3201~3202）に記憶されている各構成要素の負荷を計算する。

【0129】

例えば、DB1__Table1~999を読み込む際のデータ転送パスの構成要素の1つであるI/Oポート1131aの1並列処理単位における受信転送速度（Gbps）の負荷は、式（1）より、 $(70\text{MB} \times 1) / (3\text{ミリ秒} \times 1 + 30\text{ミリ秒} + 10\text{ミリ秒} \times 1) = 1.6\text{Gbps}$ であると計算できる。そして、要求リソース量計算プログラム1000は、例えば、I/Oポート1131aの受信転送速度の要求リソース量を「1.6Gbps」と記憶する（ステップS1007）。

【0130】

<最大並列数計算処理>

最大並列数計算処理は、最大並列数計算プログラム1100をCPU211が実行することにより行われる処理であり、空きリソース量計算処理で計算した各構成要素の空きリソース量と、要求リソース量計算処理で計算した各構成要素に対するジョブの要求リソース量と、に基づいて、各構成要素がボトルネックにならない、ジョブの最大並列数を計算する処理である。

【0131】

図12は、第1実施形態に係る最大並列数計算処理のフローチャートである。

【0132】

最大並列数計算処理は、例えば、空きリソース量計算処理と、要求リソース量計算処理との完了を検知した時に開始される。

【0133】

ステップS1101において、最大並列数計算プログラム1100（厳密には、最大並列数計算プログラム1100を実行するCPU211）は、空きリソース量計算処理で計算して記憶した登録されたジョブのパス上の各構成要素の空きリソース量と、要求リソース量計算処理で計算して記憶した、1並列処理単位あたりの各構成要素に対する要求リソース量とを取得する。

【0134】

ステップS1102において、最大並列数計算プログラム1100は、各構成要素の空きリソース量と、1並列処理単位あたりの各構成要素に対する要求リソース量とに基づいて、ジョブの並列数を徐々に増やした場合に、最も少ない並列数でリソース量の空きがなくなる構成要素を特定し、その構成要素のリソース量の空きがならない場合の最大の並列数を計算し、その並列数をジョブの最大並列数とする。例えば、最大並列数は、パス上のそれぞれの構成要素についての（空きリソース量 / 要求リソース量）を求め、算出されたそれらの値の中の最小値を超えない最大の整数値としてもよい。

【0135】

ステップS1103において、最大並列数計算プログラム1100は、ステップS1102で計算した最大並列数を表示部5100に出力する。

【0136】

最大並列数計算処理の具体例は以下のとおりである。例えば、ステップS1101で、最大並列数計算プログラム1100は、空きリソース量計算処理で計算したパスの各構成要素の空きリソース量と、要求リソース量計算処理で計算した登録ジョブのパスの各構成

10

20

30

40

50

要素に対する要求リソース量とを受信する。ステップ S 1 1 0 2 で、最大並列数計算プログラム 1 1 0 0 は、例えば、I / O ポート 1 3 1 a の空きリソース量が 8 . 0 G b p s であり、要求リソース量が 1 . 6 G b p s である場合、I / O ポート 1 3 1 a が許容する最大の並列数は、 $8 . 0 / 1 . 6 = 5$ と算出する。並列数 5 は、登録ジョブの並列数を 5 まで許容することを意味する。さらに、最大並列数計算プログラム 1 1 0 0 は、パスの他の構成要素についても同様に計算する。ここで、最も小さい並列数を許容する構成要素がボトルネックとなり、その構成要素の許容する並列数がデータ分析基盤における最大並列数となる。そこで、最大並列数計算プログラム 1 1 0 0 は、このようなボトルネックとなる構成要素の許容する並列数から最大並列数を算出する。ステップ S 1 1 0 3 では、最大並列数計算プログラム 1 1 0 0 は、表示部 5 1 2 0 0 に、例えば、最大並列数を含む出力画面（例えば、図 1 3 参照）を表示する。

10

【 0 1 3 7 】

< 出力画面 5 1 2 1 0 >

図 1 3 は、第 1 実施形態に係る出力画面の一例を示す図である。図 1 3 に示す出力画面は、G U I で実装した場合の一例を示す。

【 0 1 3 8 】

出力画面 5 1 2 1 0 は、登録されたジョブのジョブ名を表示する表示領域 5 1 2 1 1 と、ジョブの推奨する並列処理数を表示する表示領域 5 1 2 1 2 とを有する。表示領域 5 1 2 1 2 には、最大並列数計算処理により算出された最大並列数が表示される。

【 0 1 3 9 】

例えば、図 1 3 の出力画面 5 1 2 1 0 は、「ジョブ A」の処理は、並列数を「20」で実行することが推奨されることを示している。

20

【 0 1 4 0 】

なお、上記実施形態においては、最大並列数を出力画面 5 1 2 1 0 に表示させるようにしていたが、例えば、管理計算機 2 0 0 は、最大並列数計算処理により算出された最大並列数でジョブを実行するようにジョブ実行サーバ 1 1 0 に対して設定を行う機能を有していてもよい。

【 0 1 4 1 】

以上に説明したように、第 1 実施形態によれば、例えば、ジョブを実行する際に、ジョブのデータ転送を行うパス上の構成要素の負荷と、最大性能値とに基づいて、各構成要素がボトルネックとならないようなジョブの最大並列数を計算できる。これにより、ジョブの処理完了時間が最も短く、かつ、パブリッククラウド環境 1 0 1 におけるコンピュータに対する課金額が小さくなる並列処理数を適切に決定することができる。

30

【 0 1 4 2 】

なお、第 1 実施形態においては、ジョブが実行するプロセスの並列数がジョブの実行時におけるどの時刻においても設定した並列数を保持することを前提とした例を示している。しかし、実際の E T L 処理においては、特にジョブの起動直後においては、起動のための処理負荷などにより、設定した並列数で実行されない場合がある。

【 0 1 4 3 】

ここで、ジョブにおける要求リソース量の変化について説明する。

40

【 0 1 4 4 】

図 1 4 は、ジョブの要求リソース量の変化を示す図である。図 1 4 (a) は、第 1 実施形態における理想的なジョブの要求リソース量の変化を示し、図 1 4 (b) は、一般的なジョブの要求リソース量の変化を示す。

【 0 1 4 5 】

例えば、第 1 実施形態では、グラフ 5 1 3 0 1 に示すように要求リソース量は理想的にジョブの起動とともにジョブの並列数だけ増え、ジョブの終了とともに減少することを想定している。しかし、実際には、グラフ 5 1 3 0 2 に示すように要求リソース量はジョブが起動されると、徐々に増加し、ジョブの終了に近づくにつれて徐々に減少する。これに対しては、グラフ 5 1 3 0 2 に示すような要求リソース量の変動をプロセス種別ごとに機

50

械学習などで学習し、並列数と要求リソース量のピーク値とを導出できるモデルを作成しておくことにより、要求リソース量のピーク値における並列数を求めることで、最大並列数を計算するようにしてもよい。例えば、プロセス種別に基づくデータ転送の応答時間と、プロセスの1並列処理単位の処理計算時間と、読込回数と、書込回数と、読み込むデータのデータ容量と読み込み単位から計算する読込データ容量と、書き込むデータのデータ容量と書き込み単位から計算する読込データ容量と、並列数とを少なくとも特徴量として持ち、要求リソース量（あるいは要求リソース量のピーク値）を算出するモデルを作成し、このモデルを使用して最大並列数を計算してよい。

【0146】

なお、第1実施形態においては、出力画面51210には、登録されたジョブの最大並列数のみを表示するようにしているが、最大並列数で実行した場合のジョブの完了予定時刻を合わせて表示してもよい。ジョブの完了予定時刻の計算方法については、後述する第2実施形態の完了予定時刻計算処理（図18）における完了予定時刻の計算方法を使用してもよい。

10

【0147】

また、第1実施形態においては、ジョブの推奨並列数を計算して表示するようにしていたが、計算した推奨並列数に基づいて、ジョブを実行するジョブ実行サーバ等のスペックを決定してもよい。例えば、1並列処理単位あたりのジョブ実行サーバ110への負荷をプロセス種別ごとに計測等して記憶しておき、推奨並列数でのジョブ実行サーバ110への負荷を計算して、この負荷を満足するようにジョブ実行サーバ110のスペック（性能）を決定してもよい。

20

【0148】

また、第1実施形態においては、1つのDBに対してRDBMSサーバ120が1台の場合を例に挙げているが、RDBMSサーバ120はクラスタリングされていてもよい。この場合、構成要素情報テーブル310に格納される構成要素の情報は、クラスタリングされたすべてのRDBMSサーバ120を1つのサーバとして扱ってもよい。例えば、ネットワークI/F153の最大性能値は、クラスタを構成するすべてのRDBMSサーバ120のそれぞれのネットワークI/F153の最大性能値を足し合わせた値としてもよい。また、RDBMSサーバ120がオートスケール機能などで自動的にサーバ台数を変更可能な場合においては、変更可能なサーバ台数の最大値における全てのサーバの構成要素の性能値を足し合わせた値を最大性能値としてよい。

30

【0149】

また、第1実施形態においては、ETL処理を実施するデータをRDBデータとしたが、データはこれに限られず、ファイル形式やオブジェクト形式のデータであってもよく、そのデータは、ファイルサーバやオブジェクトストレージに保存されて管理されていてもよい。

【0150】

また、第1実施形態においては、パブリッククラウド環境101と、オンプレミス環境102とを含むハイブリッドクラウド環境においてボトルネックになりうる主な構成要素が、データ転送にかかわる構成要素であるため、データ転送に関わる構成要素の転送速度の最大性能値と、要求負荷とに基づいて最大並列数を導出するようにしていたが、例えば、サーバのCPUの負荷や、ストレージ装置のプロセッサの負荷などを含めて最大並列数を計算してもよい。また、監視メトリック種別としては、転送速度（例えば、1秒間の転送データ量）とした例を示していたが、これに限られず、IOPS（1秒間のIO数）であってもよい。さらに、負荷や、最大性能値が計算できるのであれば、他の構成要素および監視メトリック種別を含めるようにしてもよい。

40

【0151】

また、第1実施形態においては、ジョブに対して最大並列数を決定するようにしていたが、データ分析基盤100がジョブの実行中にプロセス単位で並列数を変更できる場合には、それぞれのプロセス単位で同様な処理を実行してプロセスごとに最大並列数を決定し

50

てもよい。この場合には、プロセスごとの最大並列数を出力画面 5 1 2 1 0 に表示させるようにしてもよい。

【 0 1 5 2 】

また、第 1 実施形態においては、登録されたジョブは、登録時に即時実行される例を示していたが、例えば、スケジューラ機能などによって、指定時刻に開始されるようにしてもよい。この場合、空きリソース量計算処理で使用する各構成要素の負荷や、要求リソース量計算処理で使用する応答時間の値は、過去の負荷情報やスケジューリングされた他のジョブの情報や応答時間から予測した、指定された時刻での予測負荷や、予測応答時間を用いて計算してもよいし、ジョブの実行直前に各処理を実行することにより最大並列数を再計算してもよい。

10

【 0 1 5 3 】

また、第 1 実施形態においては、データ分析者が利用できるように分類、分割されたデータ群を識別する識別子と、それらのデータの格納先の識別子とを同じものとしていたが、本発明はこれに限られず、データ群を識別する識別子と、格納先の識別子とを別の識別子とし、それらに対応付けて管理するようにし、対応関係を特定するようにしてもよい。

【 0 1 5 4 】

また、第 1 実施形態においては、要求リソース量計算処理は、最大並列数計算処理の直前に実行される例を示しているが、例えば、データの読み書きの応答時間に変動が少ない環境の場合は、ジョブの登録時に要求リソース量計算処理を実行して、1 並列処理単位あたりの要求リソース量を記憶するようにし、その後、記憶した要求リソース量を用いて最大並列数計算処理を実行するようにしてもよい。

20

【 0 1 5 5 】

また、第 1 実施形態においては、ジョブの並列数はジョブ実行中に変更できないことを前提として説明していたが、プロセス実行中に並列数を変更できるようにしてもよく、この場合には、例えば、他のジョブ、あるいは他のプロセスが完了した時点で、上記した空きリソース量計算処理、最大並列数計算処理等を実行することにより、最大並列数を計算するようにしてもよい。

【 0 1 5 6 】

また、第 1 実施形態においては、1 並列処理単位について 1 単位のデータを読み込むことを前提としているが、ETLプラットフォームによっては、1つのプロセスが実行される際に、プロセス実行開始前に並列化された各ジョブ実行サーバが割り当てられたデータをまとめて読み込むものがある。このようなプラットフォームに対応するために、1 並列処理単位あたりの要求リソース量を計算する際に、このようなプラットフォームの特徴を考慮して計算してもよい。例えば、1 並列処理単位あたりの I/O ポート 1 3 1 a の送受信転送速度の負荷を計算する際には、プロセスが実行する「1 回の処理計算時間」を除いて計算してよい。

30

【 0 1 5 7 】

第 2 実施形態

次に、第 2 実施形態に係るデータ分析基盤管理システムについて説明する。以下の説明では、第 1 実施形態との差異を中心に説明し、同等の構成要素や、同等の機能を持つプログラム、同等の項目を持つテーブルについては、同一の符号を用い、記載を省略又は簡略する。

40

【 0 1 5 8 】

第 1 実施形態に係るデータ分析基盤管理システム 1 では、データ分析基盤 1 0 0 の負荷に応じて、ETL 処理のジョブの並列数を決定した。第 1 実施形態においては、ある ETL 処理のジョブを新たに実行する際に、他のジョブによってデータ分析基盤の 1 つの構成要素のリソースが最大まで利用されていた場合、新たなジョブの並列数は極端に少なくなる（例えば、1 になる）場合がある。しかし、このような少ない並列数で実行するよりも、他のジョブの完了を待った後に、データ分析基盤 1 0 0 のリソースを最大限に利用した方が早く処理が終わる場合がある。

50

【 0 1 5 9 】

そこで、第2実施形態に係るデータ分析基盤管理システムは、或る登録されたジョブの実行にあたって、他のE T L処理のジョブの完了時刻を予測し、登録されたジョブを指定時刻に実行するか、他のジョブの完了を待って実行するかを判定するようにしている。

【 0 1 6 0 】

第2実施形態に係る管理計算機200は、登録ジョブ情報記憶部800が登録ジョブ情報テーブル810に代えて登録ジョブ情報テーブル850を格納するとともに、管理計算機200が最大並列数計算プログラム1100に代えて最大並列数計算プログラム1110を記憶する。

【 0 1 6 1 】

<登録ジョブ情報記憶部800>

登録ジョブ情報記憶部800は、登録ジョブ情報テーブル850を記憶する。

【 0 1 6 2 】

図15は、第2実施形態に係る登録ジョブ情報テーブルの構成図である。なお、登録ジョブ情報テーブル810と同様なフィールドには、同一の符号を付し、説明を省略する。

【 0 1 6 3 】

登録ジョブ情報テーブル850のエントリは、データ分析者によって登録されたE T L処理のジョブの情報を格納すべく、ジョブID811と、プロセスID812と、プロセス種別ID813と、パラメータ814と、データソース815と、アウトプット816と、開始時刻851と、完了予定時刻852と、並列数853とのフィールドを有する。開始時刻851には、ジョブの実行を開始する時刻（開始時刻）が格納される。完了予定時刻852には、ジョブの完了時刻を予測した値（完了予定時刻）が格納される。並列数853には、実行中あるいは実行予定のジョブでの並列数が格納される。

【 0 1 6 4 】

例えば、登録ジョブ情報テーブル850のエントリ14001は、登録されたジョブAは2020年1月1日0時に開始され、完了予定時刻は2020年1月1日12時であり、ジョブは、並列数が20で実行されることを示す。

【 0 1 6 5 】

<最大並列数計算処理>

最大並列数計算処理は、管理計算機200の最大並列数計算プログラム1110をCPU211が実行することにより行われる処理であり、登録されたジョブを各々の他のジョブの完了予定時刻後に開始した場合の最大並列数と完了予定時刻を計算し、最も完了予定時刻が早くなる開始時刻と完了予定時刻と並列数とを計算する処理を含んでいる。

【 0 1 6 6 】

図16は、第2実施形態に係る最大並列数計算処理のフローチャートである。

【 0 1 6 7 】

最大並列数計算処理は、例えば、空きリソース量計算処理と、要求リソース量計算処理との完了が検知された時に実行される。

【 0 1 6 8 】

ステップS1501において、最大並列数計算プログラム1110は、第1実施形態の最大並列数計算処理（図12参照）を実行する。

【 0 1 6 9 】

ステップS1502において、最大並列数計算プログラム1110は、完了予定時刻計算処理（図18参照）を実行することにより、空きリソース量計算処理のステップS901で取得した登録ジョブについて、或る開始時刻（例えば、データ分析者が指定した時刻、現在時刻、あるいは、任意の時刻）に対し、ステップS1102で計算した並列数で実行した場合の完了予定時刻を予測して、記憶する。

【 0 1 7 0 】

ステップS1503において、最大並列数計算プログラム1110は、他のジョブ情報を登録ジョブ情報テーブル850から取得する。なお、ここで、取得する他のジョブ情報

10

20

30

40

50

は、同じ期間に実行されるジョブに限定してよい。同じ期間とは、例えば、登録ジョブの開始時刻から完了予定時刻が示す期間が、他のジョブの開始時刻から完了予定時刻が示す期間と重複する場合を示す。また、ここで取得するジョブは、登録ジョブとデータ転送のパスが重複するジョブに限定してよい。

【0171】

ステップS1504において、最大並列数計算プログラム1110は、ステップS1503で取得した他のジョブの情報をその完了予定時刻順にソートし、キューに記憶する。キューは、例えば、メモリ212上に記憶されていてもよい。

【0172】

ステップS1505において、最大並列数計算プログラム1110は、キューからジョブ情報を1つ取得する。ここで、取得したジョブ情報のジョブを対象ジョブということとする。

10

【0173】

ステップS1506において、最大並列数計算プログラム1110は、対象ジョブの実行時の要求リソース量を計算する。例えば、最大並列数計算プログラム1110は、要求リソース量計算プログラム1000を呼び出して対象ジョブの1並列処理単位あたりの要求リソース量を計算し、登録ジョブ情報テーブル850からジョブ情報の並列数853の並列数を取得し、以下の式(4)で要求リソース量を計算する。

$$\text{要求リソース量} = (\text{1並列処理単位あたりの要求リソース量}) \times (\text{並列数}) \cdots (4)$$

なお、本実施形態では1つのジョブに対して要求リソース量はジョブ全体で一定としているが、例えば、順次実行されるプロセスを複数持つジョブの場合は、プロセスごとに要求リソース量を計算してもよい。

20

【0174】

ステップS1507において、最大並列数計算プログラム1110は、対象ジョブの実行時の空きリソース量を計算する。例えば、最大並列数計算プログラム1110は、空きリソース量計算プログラム900を呼び出して空きリソース量を計算する。なお、対象ジョブの開始時刻が現在時刻より後の場合は、開始時刻以降に実行される他のジョブの要求リソース量を計算して負荷とし、空きリソース量を計算してもよく、あるいは、開始時刻以降の負荷を公知の負荷予測アルゴリズムを用いて予測し、予測した負荷を用いて、空きリソース量を計算してもよい。

30

【0175】

ステップS1508において、最大並列数計算プログラム1110は、ステップS1507で計算した対象ジョブ実行時の空きリソース量からステップS1506で計算した対象ジョブの要求リソース量を減算して、対象ジョブ完了時の各構成要素の空きリソース量を計算する。

【0176】

ステップS1509において、最大並列数計算プログラム1110は、対象ジョブ完了後に登録ジョブを開始した時の最大並列数を計算する。最大並列数は、ステップS1508の対象ジョブ完了時の空きリソース量と、要求リソース量計算プログラム900で計算した登録ジョブの1並列処理単位あたりの要求リソース量とに基づいて、第1実施形態の最大並列数計算処理と同様な処理により計算してもよい。

40

【0177】

ステップS1510において、最大並列数計算プログラム1110は、完了予定時刻計算処理(図18参照)により、対象ジョブの完了予定時刻を登録ジョブの開始時刻とした時の完了予定時刻を計算する。

【0178】

ステップS1511において、最大並列数計算プログラム1110は、ステップS1510で計算した完了予定時刻が、ステップS1502で記憶している完了予定時刻より早くなるか否かを判定する。この判定の結果が真の場合(S1511: YES)、最大並列数計算プログラム1110は、処理をステップS1512へ進め、この判定の結果が偽の

50

場合 (S 1 5 1 1 : N O)、処理をステップ S 1 5 1 3 へ進める。

【 0 1 7 9 】

ステップ S 1 5 1 2 において、最大並列数計算プログラム 1 1 1 0 は、ステップ S 1 5 0 2 で記憶した登録ジョブの完了予定時刻を、ステップ S 1 5 1 0 で計算した完了予定時刻に更新する。

【 0 1 8 0 】

ステップ S 1 5 1 3 において、最大並列数計算プログラム 1 1 1 0 は、キューが空か否かを判定する。この判定の結果が真の場合 (S 1 5 1 3 : Y E S)、最大並列数計算プログラム 1 1 1 0 は、処理をステップ S 1 5 1 4 へ進め、この判定の結果が偽の場合 (S 1 5 1 3 : N O)、最大並列数計算プログラム 1 1 1 0 は、処理をステップ S 1 5 0 5 へ進める。

10

【 0 1 8 1 】

ステップ S 1 5 1 4 において、最大並列数計算プログラム 1 1 1 0 は、ステップ S 1 5 0 2、あるいは、ステップ S 1 5 1 2 で記憶した登録ジョブの完了予定時刻と、その完了予定時刻を記憶した時の最大並列数と、その完了予定時刻を記憶した時にステップ S 1 5 0 2 あるいは S 1 5 1 0 で用いた開始時刻との組を含む出力画面 5 1 2 2 0 (図 1 9 参照) を表示部 5 1 2 0 0 に出力する。

【 0 1 8 2 】

ここで、最大並列数計算処理における S 1 5 0 2 から S 1 5 1 4 の処理の本質について説明する。

20

【 0 1 8 3 】

図 1 7 は、第 2 実施形態に係る最大並列数計算処理の本質を説明する図である。

【 0 1 8 4 】

最大並列数計算プログラム 1 1 1 0 によるステップ S 1 5 0 2 からステップ S 1 5 1 4 によって、開始時刻と完了予定時刻と並列数とを出力する処理は、図 1 7 に示すように、他のジョブ (ジョブ B、ジョブ C) による各構成要素の負荷と、ジョブの実行期間とが成す矩形のうち、データ分析基盤 1 0 0 の各構成要素の最大性能値を超えない、かつ、完了予定時刻が最も早くなる登録ジョブの実行期間を示す矩形 (例えば、図 1 7 のグラフ 5 1 6 0 0) の場所を探索する処理に相当する。なお、登録ジョブの実行期間を示す矩形の場所を探索する処理としては、上記に限られず、例えば、時刻に対する任意の各区間ごとに登録ジョブを実行した場合の最大並列数と開始時刻と完了予定時刻とを計算し、最適な区間を導出するようにしてもよい。

30

【 0 1 8 5 】

なお、第 2 実施形態では、登録されたジョブの開始時刻より後、かつ、完了予定時刻より前に開始される他のジョブについては説明していないが、それらのジョブの負荷を予測して登録ジョブの最大並列数を計算してもよい。

【 0 1 8 6 】

また、ジョブの優先度を設定し、他のジョブの優先度が低ければ、それらのジョブの負荷を含めずに最大並列数を計算するようにしてもよい。

【 0 1 8 7 】

また、第 2 実施形態では、図 1 4 のグラフ 5 1 3 0 1 に示すように要求リソース量は理想的にジョブの起動とともにジョブの並列数だけ増え、ジョブの終了とともに減少する想定での計算を例示している。しかし、実際には図 1 4 のグラフ 5 1 3 0 2 に示すように要求リソース量は徐々に増加し、徐々に減少する。ここで、グラフ 5 1 3 0 1 のように、ジョブ起動からジョブの終了までの間、最大の要求リソース量とすることのできる並列数ではなく、その並列数よりも少ない並列数 (所定の割合 (例えば、20%) だけ少ない並列数をジョブの最大並列数として設定するようにしてもよい。このようにすると、最大の処理効率は幾分低下するが、ジョブの起動直後やジョブの終了直前において、利用されないのに確保されてしまう無駄なリソース量を低減することができる。

40

【 0 1 8 8 】

50

< 完了予定時刻計算処理 >

完了予定時刻計算処理は、最大並列数計算プログラム 1110 を CPU 211 が実行することにより行われる処理であり、指定された開始時刻と並列数とに対し、登録ジョブの完了予定時刻を計算する処理である。

【0189】

図 18 は、第 2 実施形態に係る完了予定時刻計算処理のフローチャートである。

【0190】

完了予定時刻計算処理は、最大並列数計算処理のステップ S 1502、ステップ S 1510 において実行される。

【0191】

ステップ S 1701 において、最大並列数計算プログラム 1110 は、指定された登録ジョブ情報と、開始時刻と、ジョブの並列数とを取得する。

【0192】

ステップ S 1701 の後、最大並列数計算プログラム 1110 は、ステップ S 1701 で取得した登録ジョブの有するすべてのプロセス ID 802 のプロセス ID を処理対象にループ 3 の処理 (ステップ S 1702 ~ S 1704) を実行する。ここで、処理対象のプロセス ID を対象プロセス ID というものとする。

【0193】

ステップ S 1702 において、最大並列数計算プログラム 1110 は、対象プロセス ID のデータソース 815、あるいは、アウトプット 816 の値が示すデータの応答時間を応答時間情報テーブル 510 から取得する。取得する応答時間は、例えば、データソース 815、あるいは、アウトプット 816 の値に関連する最新の応答時間であってもよく、また、完了予定時刻計算処理が開始されてから計測した応答時間であってもよい。

【0194】

ステップ S 1703 において、最大並列数計算プログラム 1110 は、対象プロセス ID に関連するプロセス種別 ID 813 のプロセス種別 ID に基づいて、プロセス種別情報テーブル 710 から関連するエントリ (1 回の処理計算時間、読込単位、書込単位、読込回数、及び書込回数) を取得する。

【0195】

ステップ S 1704 において、最大並列数計算プログラム 1110 は、ステップ S 1702 で取得した応答時間と、ステップ S 1703 で取得したエントリの 1 回の処理計算時間、読込単位、書込単位、読込回数、及び書込回数と、に基づいて、プロセスの 1 並列処理単位あたりの処理時間を計算する。例えば、以下の式 (5) でプロセスの 1 並列処理単位あたりの処理時間を計算する。

【0196】

プロセスの 1 並列処理単位あたりの処理時間

$$= (\text{読込データの応答時間} \times \text{読込回数} + 1 \text{ 回の処理計算時間} + \text{書込データの応答時間} \times \text{書込回数}) \cdots (5)$$

【0197】

ステップ S 1705 において、最大並列数計算プログラム 1110 は、ループ 3 の処理 (S 1702 ~ S 1704) で計算した各プロセスの 1 並列処理単位あたりの処理時間と、開始時刻と、並列数とに基づいて、ジョブの完了予定時刻を計算する。例えば、以下の式 (6) で完了予定時刻を計算する。

完了予定時刻 = 開始時刻 + ジョブの処理時間 $\cdots (6)$

なお、ジョブの処理時間は、以下の式 (7) により計算される。

ジョブの処理時間

$$= (\text{プロセス}) \{ (\text{プロセスの 1 並列処理単位あたりの処理時間} \times \text{読込データ数}) / (\text{並列数} \times \text{読込回数}) \} \cdots (7)$$

また、読込データ数は、プロセス種別テーブル 710 の読込単位 713 の読込単位と、登録ジョブ情報テーブル 850 のデータソース 815 の値とに基づいて計算できる、デー

10

20

30

40

50

タをストレージ装置から転送する回数であってよい。

【0198】

完了予定時刻計算処理の具体例は以下のとおりである。例えば、最大並列数計算プログラム1110は、登録ジョブ情報テーブル810のエントリ8001が示すジョブAと、開始時刻2020年1月1日12:00と、並列数5とを入力としてステップS1701において取得し、ループ3の処理(ステップS1702~ステップS1704)を繰り返す。ループ3の処理において、最大並列数計算プログラム1110は、エントリ8001のプロセスID「プロセスa」に対して、データソース815が示すデータのデータ転送の応答時間3ミリ秒と、アウトプット816が示す「DB3__Table1」のデータ転送の応答時間10ミリ秒とを応答時間情報テーブル510から取得する(ステップS1703)。ステップS1703において、最大並列数計算プログラム1110は、プロセス種別情報テーブル710からプロセス種別IDが「Table__to__Table__Extract__Column」であるエントリ7001を取得し、1回の処理計算時間「30ミリ秒」、読込回数「1」、書込回数「1」、読込単位「1テーブル」、書込単位「1テーブル」、を取得する。ステップS1704において、最大並列数計算プログラム1110は、1並列処理単位あたりの処理時間 = (3ミリ秒 × 1 + 30ミリ秒 + 10ミリ秒 × 1) = 43ミリ秒を計算する。

10

【0199】

ループ3の処理の後にステップS1705において、最大並列数計算プログラム1110は、読込単位が1テーブルであり、データソースがテーブル1~999であることから、読込データ数は999個であり、ジョブの処理時間 = (43ミリ秒 × 999) / (5 × 1) = 8.5秒と計算する。

20

【0200】

なお、第2実施形態では、ステップS1705で計算するジョブの処理時間を、各プロセスにかかる処理時間のみを計算する例を挙げていたが、実際にはジョブ実行時にジョブの起動時間や前処理等の時間も必要となるため、それらの時間を加えて計算してもよい。

【0201】

また、第2実施形態においては、ジョブが実行するプロセスの並列数がどの時刻においても設定した並列数を保持する前提の計算を例示している。しかし、実際のETL処理においては、特にジョブの起動直後においては、起動のための処理負荷などにより、指定した並列数で実行されない場合がある。すなわち、実際には図14のグラフ51302に示すように要求リソース量は徐々に増加し、徐々に減少し、それに応じてジョブの処理時間も変動する。そこで、図14のグラフ51302に示すような処理時間の変動をプロセス種別ごとに機械学習などで学習し、ジョブの処理時間を導出できるモデルを作成しておき、完了予定時刻計算処理においてはこのモデルを使用することで完了予定時間を計算するようにしてもよい。例えば、プロセス種別に基づくデータ転送の応答時間と、プロセスの1並列処理単位の処理計算時間と、読込回数と、書込回数と、データソースと読み込み単位から計算する読込データ数と、並列数と、を少なくとも特徴量として持ち、ジョブの処理時間を算出するモデルを作成し、このモデルによりジョブの処理時間を計算してもよい。

30

【0202】

<出力画面51220>

図19は、第2実施形態に係る出力画面の一例を示す図である。図19に示す出力画面は、GUIで実装した場合の一例を示す。なお、第1実施形態に係る出力画面51210と同様な部分については、同一符号を付している。

40

【0203】

出力画面51220は、登録されたジョブのジョブ名を表示する表示領域51211と、ジョブの推奨する並列処理数を表示する表示領域51212と、登録ジョブに対して推奨する奨励開始時刻を表示する表示領域51213と、登録ジョブの完了予定時刻を表示する表示領域51214とを有する。

【0204】

50

例えば、図 19 の出力画面 5 1 2 2 0 は、「ジョブ A」の処理は並列数を 20 で実行し、開始時刻は 2020 年 1 月 2 日 0 時、ジョブの完了予定時刻は 2020 年 1 月 2 日 1 時であることを示している。

【0205】

なお、上記実施形態においては、最大並列数を出力画面 5 1 2 2 0 に表示させるようにしていたが、例えば、管理計算機 200 は、最大並列数計算処理により算出された最大並列数でジョブを実行するようにジョブ実行サーバ 110 に対して設定を行う機能を有していてもよい。

【0206】

また、出力画面 5 1 2 2 0 に、ステップ S 1509 ~ S 1510 で計算した値を用いて、他のジョブの完了を待機しなかった場合の開始時刻と完了予定時刻と並列数との組を、他のジョブの完了予定時刻ごとに表示してもよい。

10

【0207】

以上に説明したように、第 2 実施形態によれば、或る登録ジョブの実行にあたって、他の ETL 処理のジョブの完了時刻に基づいて、登録ジョブを指定時刻に実行するか、他のジョブの完了を待って実行するかを判定し、最も早くジョブを完了できる開始時刻を決定できる。

【0208】

なお、第 2 実施形態では、事前に計測されたプロセスの処理時間や応答時間を用いてジョブの完了予定時刻を計算しているが、ジョブが既に実行されている場合は、ジョブの実行時間と進捗とを計測し、完了予定時刻を予測してもよい。

20

【0209】

第 3 実施形態

次に、第 3 実施形態に係るデータ分析基盤管理システムについて説明する。以下の説明では、第 1、第 2 実施形態との差異を中心に説明し、同等の構成要素や、同等の機能を持つプログラム、同等の項目を持つテーブルについては、同一の符号を用い、記載を省略又は簡略する。

【0210】

第 1、第 2 実施形態に係るデータ分析基盤管理システムでは、データ分析基盤 100 の負荷や他の ETL 処理のジョブの完了予定時刻に応じて、ジョブの完了が最短となるジョブの並列数あるいはジョブの開始時刻を決定した。しかし、データ分析者がジョブの完了時刻に対して特定の期限を有する場合に、第 1、第 2 実施形態の方法では、他のジョブによってデータ分析基盤のリソースが使用されることによって、期限内にジョブを完了できない場合がある。

30

【0211】

そこで、第 3 実施形態に係るデータ分析基盤管理システムでは、データ分析者にジョブ完了の許容期限を設定させ、或る登録ジョブの実行にあたって、ジョブの完了予定時刻を予測し、完了予定時刻が許容期限を超える場合は、他のジョブの許容期限の範囲内で他のジョブの並列数を変更する例について説明する。

【0212】

第 3 実施形態に係る管理計算機 200 は、入力画面 5 1 1 1 0 に代えて入力画面 5 1 1 2 0 を表示させるようにし、第 2 実施形態に係る登録ジョブ情報記憶部 800 が登録ジョブ情報テーブル 850 に代えて登録ジョブ情報テーブル 860 を格納し、管理計算機 200 が最大並列数計算プログラム 1110 に代えて最大並列数計算プログラム 1120 を記憶する。

40

【0213】

<入力画面 5 1 1 2 0 >

図 20 は、第 3 実施形態に係る入力画面の一例を示す図である。図 20 に示す入力画面は、GUI で実装した場合の一例を示す。なお、第 1 実施形態に係る入力画面 5 1 1 1 0 と同様な部分については、同一符号を付している。

50

【0214】

入力画面51120の入力領域51111は、データノード51112と、プロセスノード51113と、アウトプットノード51114と、ジョブ完了許容期限51115と、を有する。ジョブ完了許容期限51115は、登録したジョブの完了時刻についてデータ分析者が許容できる期限を定義できる領域である。この入力画面51120に対して入力された情報に基づいて、登録ジョブ情報テーブル860に、実行予定、あるいは実行中のジョブの完了許容期限に関する情報が格納される。

【0215】

例えば、入力画面51120は、ジョブAの完了時刻の許容期限は2020年1月2日4時であることを示している。

【0216】

<登録ジョブ情報記憶部800>

登録ジョブ情報記憶部800は、登録ジョブ情報テーブル860を記憶する。

【0217】

図21は、第3実施形態に係る登録ジョブ情報テーブルの構成図である。なお、登録ジョブ情報テーブル850と同様なフィールドには、同一の符号を付し、説明を省略する。

【0218】

登録ジョブ情報テーブル860のエントリは、データ分析者によって登録されたETL処理のジョブの情報を格納すべく、ジョブID811と、プロセスID812と、プロセス種別ID813と、パラメータ814と、データソース815と、アウトプット816と、開始時刻851と、完了予定時刻852と、並列数853と、要求リソース量861と、許容期限862と、最小並列数863と、最小要求リソース量864と、のフィールドを有する。

【0219】

要求リソース量861には、並列数853に格納された並列数でジョブを実行した時の要求リソース量が格納される。許容期限862には、データ分析者によって入力されたジョブの完了許容期限が格納される。最小並列数863には、完了許容期限を満たすために最低限必要な並列数が格納される。最小要求リソース量864には、最小並列数で実行した場合にデータ分析基盤100の各構成要素に対する要求リソース量(最小要求リソース量)が格納される。

【0220】

例えば、登録ジョブ情報テーブル860のエントリ19001は、登録されたジョブAが並列数20で実行され、その時の要求リソース量は、例えば、ネットワークI/F153の受信転送速度に対しては2Gbpsであることを示す。さらに、少なくとも2020年1月2日4時までにはジョブを完了することが要求され、それに対して必要なジョブの最小並列数は10であり、最小並列数で実行する時の要求リソース量は、例えばネットワークI/F153の受信転送速度に対しては1Gbpsであることを示している。

【0221】

なお、最小並列数と最小要求リソース量とは、ジョブの完了許容期限が設定された時点で、計算されてもよい。例えば、完了予定時刻計算処理を実行し、開始時刻は指定された値を入力し、並列数は任意の値に変更しながら、完了予定時刻計算処理により出力される完了予定時刻が完了許容期限の値に最も近くなる値を探索することにより導出してもよい。また、要求リソース量と、最小要求リソース量とは、要求リソース量計算処理によって計算された1並列処理単位あたりの要求リソース量と、並列数853の並列数、あるいは、最小並列数863の最小並列数に基づいて計算してもよい。

【0222】

<最大並列数計算処理>

最大並列数計算処理は、管理計算機200の最大並列数計算プログラム1120をCPU211が実行することにより行われる処理であり、登録ジョブが現状の空きリソース量に対して完了許容期限を満たさない場合に、登録ジョブと他のジョブの完了許容期限をす

10

20

30

40

50

べて満たす、他のジョブの並列数（あるいは使用リソース量）を探索する処理を更に含んでいる。

【0223】

図22は、第3実施形態に係る最大並列数計算処理のフローチャートである。

【0224】

最大並列数計算処理は、例えば、空きリソース量計算処理と、要求リソース量計算処理との完了が検知された時に実行される。

【0225】

ステップS2001において、最大並列数計算プログラム1120は、第2実施形態の最大並列数計算処理（図16参照）を実行する。

10

【0226】

ステップS2002において、最大並列数計算プログラム1120は、空きリソース量計算処理のステップS901で取得した登録ジョブについて、登録ジョブ情報テーブル860から完了許容期限を取得する。

【0227】

ステップS2003において、最大並列数計算プログラム1120は、ステップS2001で導出した完了予定時刻が完了許容期限より遅いか否かを判定する。この判定の結果が真の場合（S2003：YES）、最大並列数計算プログラム1120は、処理をステップS2004へ進め、この判定の結果が偽の場合（S2003：NO）、最大並列数計算プログラム1120は、処理をステップS2012へ進める。

20

【0228】

ステップS2004において、最大並列数計算プログラム1120は、空きリソース量計算処理で計算した、登録ジョブのデータ転送パス上の構成要素の空きリソース量を記憶する。

【0229】

ステップS2005において、最大並列数計算プログラム1120は、他のジョブ情報を登録ジョブ情報テーブル860から取得してキューに記憶する。なお、ここで取得する他のジョブ情報は、同じ期間に実行されるジョブに限定してよい。同じ期間とは、例えば、登録ジョブの開始時刻から完了許容期限が示す期間が、他のジョブの開始時刻から完了許容期限が示す期間と重複する場合を示す。また、ここで取得するジョブは、登録ジョブとデータ転送のパスが重複するジョブに限定してよい。

30

【0230】

ステップS2006において、最大並列数計算プログラム1120は、ステップS2005のキューが空か否かを判定する。この判定の結果が真の場合（S2006：YES）、最大並列数計算プログラム1120は、処理をステップS2013へ進め、この判定の結果が偽の場合（S2006：NO）、最大並列数計算プログラム1120は、処理をステップS2007へ進める。

【0231】

ステップS2007において、最大並列数計算プログラム1120は、ステップS2005のキューからジョブ情報を1つ取得する。ここで、取得したジョブ情報のジョブを対象ジョブということとする。

40

【0232】

ステップS2008において、最大並列数計算プログラム1120は、ステップS2004で記憶した空きリソース量から対象ジョブが最小並列数に変更された場合の各構成要素の空きリソース量を計算する。対象ジョブの現在の並列数設定値から最小並列数に減らす場合は、削減可能なリソース量は、（要求リソース量 - 最小要求リソース量）で計算できる。そして空きリソース量にその値を加算することで最小並列数に変更された場合の空きリソース量を計算できる。

【0233】

ステップS2009において、最大並列数計算プログラム1120は、ステップS20

50

04で記憶した空きリソース量を更新し、対象ジョブの情報(対象ジョブ情報)を記憶する。

【0234】

ステップS2010において、最大並列数計算プログラム1120は、登録ジョブの要求リソース量861の要求リソース量と、ステップS2009で更新した空きリソース量とを比較し、空きリソース量が登録ジョブの要求リソース量を満たすか否かを判定する。この判定の結果が真の場合(S2010: YES)、最大並列数計算プログラム1120は、処理をステップS2011へ進め、この判定の結果が偽の場合(S2010: NO)、並列数計算プログラム1120は、処理をステップS2006へ進める。

【0235】

ステップS2011において、最大並列数計算プログラム1120は、ステップS2009で記憶した他のジョブ(対象ジョブ)の識別情報と、最小並列数との組を出力する。例えば、最大並列数計算プログラム1120は、出力画面51220を表示し、他のジョブの並列数の変更をデータ分析者に対して要求してもよく、あるいは、ジョブ実行サーバ110に並列数の変更をリクエストするようにしてもよい。

【0236】

ステップS2012において、最大並列数計算プログラム1120は、登録ジョブの最大並列数(この例では、最小並列数と同値)と、開始時刻と、完了予定時刻とを出力する。例えば、最大並列数計算プログラム1120は、出力画面51220を表示し、登録ジョブの実行の設定をデータ分析者に対して要求してもよく、あるいは、ジョブ実行サーバ110に登録ジョブの実行の設定をリクエストするようにしてもよい。

【0237】

ステップS2013において、最大並列数計算プログラム1120は、完了予定時刻が完了許容時刻より遅くなることを、出力する。例えば、最大並列数計算プログラム1120は、完了予定時刻が完了許容時刻より遅くなることを出力画面51220に表示する。

【0238】

以上に説明したように、第3実施形態によれば、登録ジョブの実行にあたって、データ分析者によって設定された完了許容期限に対し、空きリソース量が不足していた場合に、完了許容期限を満たすために並列数を減らす他のジョブの組み合わせを特定できる。

【0239】

なお、第3実施形態においては、最大並列数計算処理において、第2実施形態の最大並列数計算処理をすべて実行するようしていたが、第2実施形態の最大並列数計算処理の開始時刻を探索する処理(ステップS1503~S1514)を実行せずに、ステップS1501~S1502のみを実行するようにしてもよい。

【0240】

また、第3実施形態においては、ジョブの実行中にジョブの並列数を変更できる場合の例を示していたが、データ分析基盤がジョブの実行中に並列数を変更できない場合は、サーバやストレージ装置やRDBMSソフトウェア等によってジョブの使用可能リソース量を変更(削減)することで登録ジョブの利用可能リソース量を増やすようにしてもよい。例えば、ストレージ装置130が特定のサーバから特定のI/Oポートに対するスループットに上限を設定する機能を有する場合には、他のジョブのI/Oポートの使用可能リソース量を減らし、それに伴って、他の構成要素への要求リソース量を減らすことができる。これにより、ジョブ実行サーバ110で実行するジョブの並列数を変更できない場合でも、登録ジョブの実行を適切に行うことができる。

【0241】

この場合には、最大並列数計算処理のステップS2008では、最大並列数計算プログラムは、最小並列数で実行した時の空きリソース量ではなく、最小要求リソース量を満たす空きリソース量(すなわち、最小並列数で実行した時の空きリソース量と同等)を計算する。また、例えば、I/Oポートのスループット上限設定機能を用いる場合、この機能では、RDBMSサーバ120とI/Oポート131との間のスループットのみ上限を設

10

20

30

40

50

定できるため、最大並列数計算処理のステップ S 2 0 0 7 では、他のジョブ情報をキューから取り出す際には、同じ R D B M S サーバ 1 2 0 にアクセスするジョブをグルーピングし、グルーピングしたジョブをまとめてキューから取得して、このグルーピングしたジョブに対して以降の処理を実行する必要がある。

【 0 2 4 2 】

なお、本発明は、上述の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲で、適宜変形して実施することが可能である。

【 0 2 4 3 】

例えば、上記実施形態において、プロセッサが行っていた処理の一部又は全部を、専用のハードウェア回路で行うようにしてもよい。また、上記実施形態におけるプログラムは、プログラムソースからインストールされてよい。プログラムソースは、プログラム配布サーバ又は記憶メディア（例えば可搬型の記憶メディア）であってもよい。

【符号の説明】

【 0 2 4 4 】

1 ... データ分析基盤管理システム、 1 0 0 ... データ分析基盤、 1 2 0 ... パブリッククラウド環境、 1 0 2 ... オンプレミス環境、 1 1 0 d , 1 1 0 e ... ジョブ実行サーバ、 1 2 0 a , 1 2 0 b , 1 2 0 c ... R D M S サーバ、 1 3 0 ... ストレージ装置、 1 5 0 a , 1 5 0 b , 1 5 0 c , 1 5 0 d , 1 5 0 e ... サーバ、 2 0 0 ... 管理計算機、 2 1 1 ... C P U、 2 1 2 ... メモリ、 2 1 3 ... ディスク

10

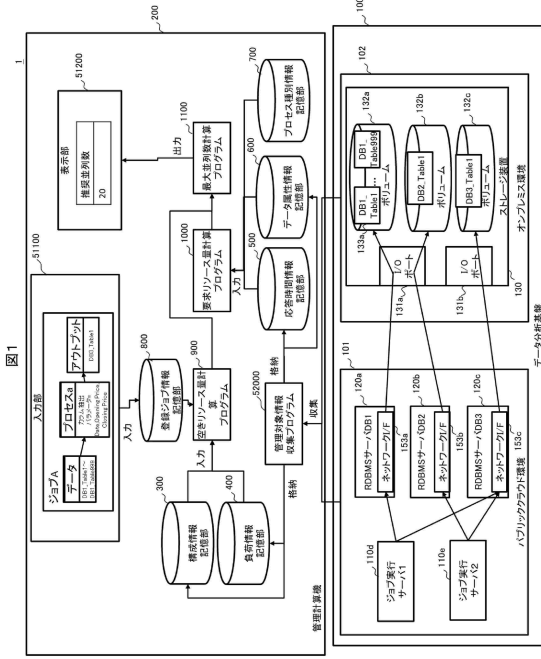
20

30

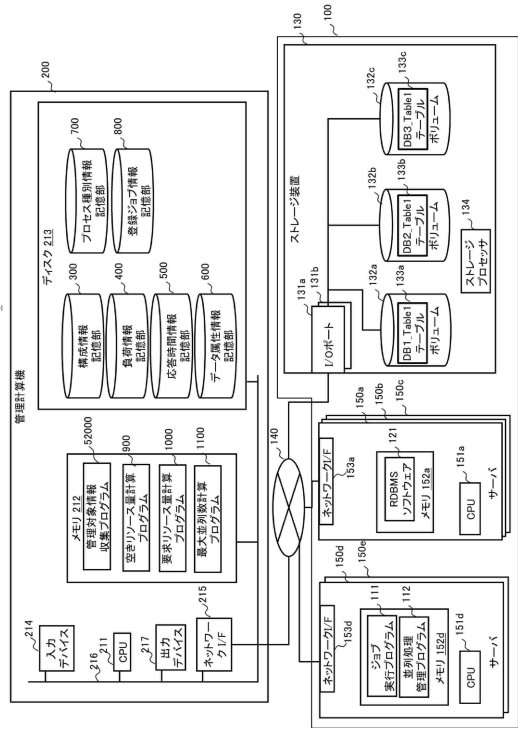
40

50

【図面】
【図 1】



【図 2】



10

20

【図 3】

図3

構成要素情報テーブル 310		バス情報テーブル 320	
310a	IOポート151a	321	データID
310b	IOポート151b	322	ネットワークID
310c	IOポート151c	323	ネットワーク140
310d	IOポート151d	324	ネットワーク131a
310e	IOポート151e	325	ネットワーク131b
310f	IOポート151f	326	ネットワーク131c
310g	IOポート151g	327	ネットワーク131d
310h	IOポート151h	328	ネットワーク131e
310i	IOポート151i	329	ネットワーク131f
310j	IOポート151j	330	ネットワーク131g
310k	IOポート151k	331	ネットワーク131h
310l	IOポート151l	332	ネットワーク131i
310m	IOポート151m	333	ネットワーク131j
310n	IOポート151n	334	ネットワーク131k
310o	IOポート151o	335	ネットワーク131l
310p	IOポート151p	336	ネットワーク131m
310q	IOポート151q	337	ネットワーク131n
310r	IOポート151r	338	ネットワーク131o
310s	IOポート151s	339	ネットワーク131p
310t	IOポート151t	340	ネットワーク131q
310u	IOポート151u	341	ネットワーク131r
310v	IOポート151v	342	ネットワーク131s
310w	IOポート151w	343	ネットワーク131t
310x	IOポート151x	344	ネットワーク131u
310y	IOポート151y	345	ネットワーク131v
310z	IOポート151z	346	ネットワーク131w
310aa	IOポート151aa	347	ネットワーク131x
310ab	IOポート151ab	348	ネットワーク131y
310ac	IOポート151ac	349	ネットワーク131z
310ad	IOポート151ad	350	ネットワーク131aa
310ae	IOポート151ae	351	ネットワーク131ab
310af	IOポート151af	352	ネットワーク131ac
310ag	IOポート151ag	353	ネットワーク131ad
310ah	IOポート151ah	354	ネットワーク131ae
310ai	IOポート151ai	355	ネットワーク131af
310aj	IOポート151aj	356	ネットワーク131ag
310ak	IOポート151ak	357	ネットワーク131ah
310al	IOポート151al	358	ネットワーク131ai
310am	IOポート151am	359	ネットワーク131aj
310an	IOポート151an	360	ネットワーク131ak
310ao	IOポート151ao	361	ネットワーク131al
310ap	IOポート151ap	362	ネットワーク131am
310aq	IOポート151aq	363	ネットワーク131an
310ar	IOポート151ar	364	ネットワーク131ao
310as	IOポート151as	365	ネットワーク131ap
310at	IOポート151at	366	ネットワーク131aq
310au	IOポート151au	367	ネットワーク131ar
310av	IOポート151av	368	ネットワーク131as
310aw	IOポート151aw	369	ネットワーク131at
310ax	IOポート151ax	370	ネットワーク131au
310ay	IOポート151ay	371	ネットワーク131av
310az	IOポート151az	372	ネットワーク131aw
310ba	IOポート151ba	373	ネットワーク131ax
310bb	IOポート151ba	374	ネットワーク131ay
310bc	IOポート151ba	375	ネットワーク131az
310bd	IOポート151ba	376	ネットワーク131ba
310be	IOポート151ba	377	ネットワーク131bb
310bf	IOポート151ba	378	ネットワーク131bc
310bg	IOポート151ba	379	ネットワーク131bd
310bh	IOポート151ba	380	ネットワーク131be
310bi	IOポート151ba	381	ネットワーク131bf
310bj	IOポート151ba	382	ネットワーク131bg
310bk	IOポート151ba	383	ネットワーク131bh
310bl	IOポート151ba	384	ネットワーク131bi
310bm	IOポート151ba	385	ネットワーク131bj
310bn	IOポート151ba	386	ネットワーク131bk
310bo	IOポート151ba	387	ネットワーク131bl
310bp	IOポート151ba	388	ネットワーク131bm
310bq	IOポート151ba	389	ネットワーク131bn
310br	IOポート151ba	390	ネットワーク131bo
310bs	IOポート151ba	391	ネットワーク131bp
310bt	IOポート151ba	392	ネットワーク131bq
310bu	IOポート151ba	393	ネットワーク131br
310bv	IOポート151ba	394	ネットワーク131bs
310bv	IOポート151ba	395	ネットワーク131bt
310bv	IOポート151ba	396	ネットワーク131bu
310bv	IOポート151ba	397	ネットワーク131bv
310bv	IOポート151ba	398	ネットワーク131bv
310bv	IOポート151ba	399	ネットワーク131bv
310bv	IOポート151ba	400	ネットワーク131bv

【図 4】

図4

IOポート131aの受信転送速度の 負荷情報テーブル 410		IOポート131aの送信転送速度の 負荷情報テーブル 420	
410a	時刻	401c	観測値
410b	2020/01/01 0:00	402b	250MB/s
410c	2020/01/01 0:01	402c	252MB/s
410d	2020/01/01 0:02	402d	243MB/s
410e	2020/01/01 0:03	402e	250MB/s
410f	...	402f	...

30

40

50

【 図 5 】

図5
応答時間情報テーブル 510

時刻	データID	処理種別	応答時間
2020/01/01:0:00	DB1	読み込み	3ms
2020/01/01:0:00	DB2_Table1	読み込み	5ms
2020/01/01:0:00	DB3_Table1	書き込み	10ms
...

【 図 6 】

図6
データ属性情報テーブル 610

データID	データ容量
DB1_Table1	50MB
DB1_Table2	120MB
...	...
DB1_Table999	20MB
DB2_Table1	100MB
...	...

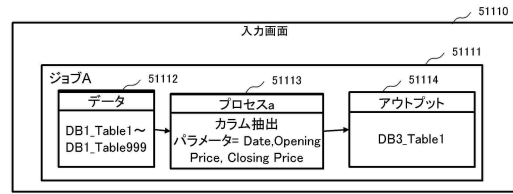
【 図 7 】

図7
プロセス種別情報テーブル 710

プロセス種別ID	1回の処理計算時間(msec)	読込単位	書込単位	読込回数	書込回数
Table_to_Table_Extract_Column	30	1テーブル	1テーブル	1	1
Table_to_Csv	20	1行	1ファイル	1	1
Csv_to_json	10	1ファイル	1ファイル	1	1
...

【 図 8 】

図8



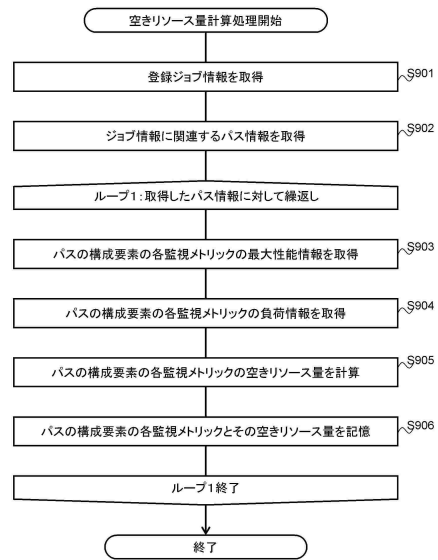
【 図 9 】

図9
登録ジョブ情報テーブル 810

ジョブID	プロセスID	プロセス種別ID	パラメータ	データソース	アウトプット
ジョブA	プロセスa	Table_to_Table_Extract_Column	Date, opening price, closing price	DB1_Table1 ~ DB1_Table999	DB3_Table1
...

【 図 10 】

図10



10

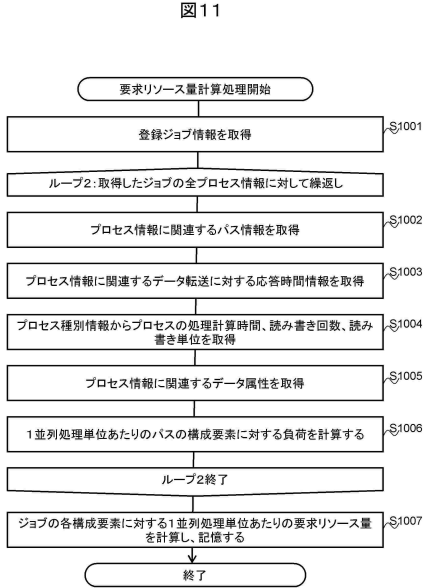
20

30

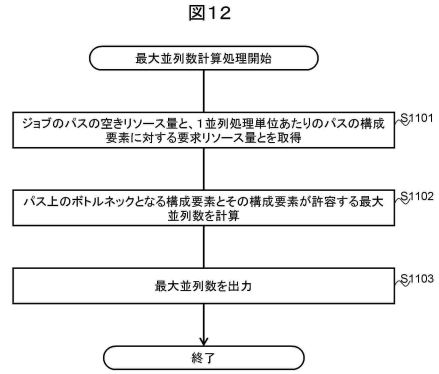
40

50

【 図 1 1 】

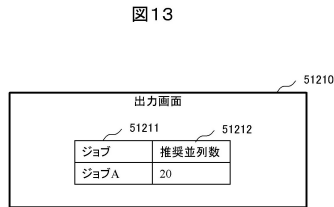


【 図 1 2 】

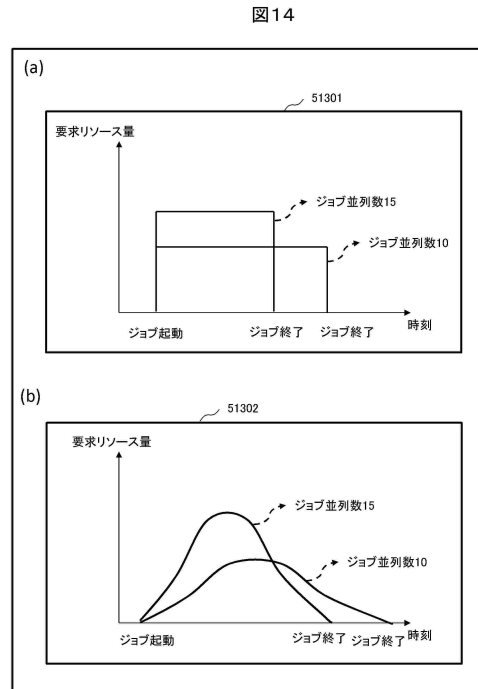


10

【 図 1 3 】



【 図 1 4 】



20

30

40

50

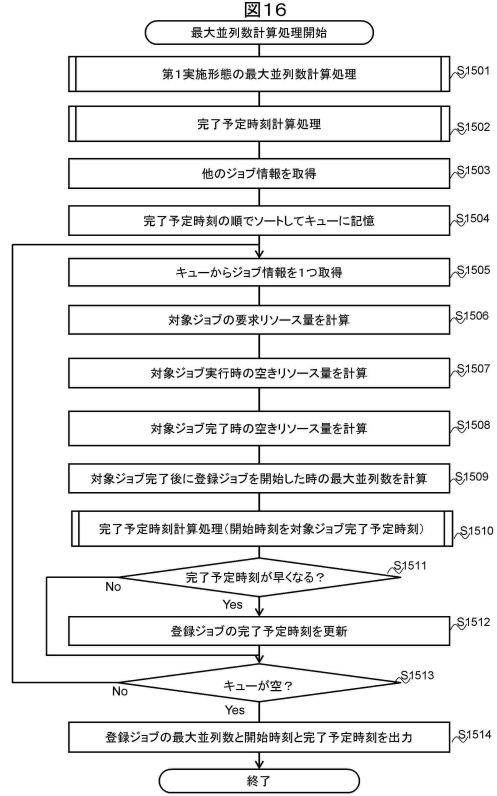
【 図 15 】

図 15

登録ジョブ情報テーブル 850

ジョブID	プロセスID	プロセス種別ID	パラメータ	データソース	アクトポイント	開始時刻	完了予定時刻	並列数
ジョブA	プロセスA	Table_Extract_Column	Date opening price, closing price	DB1_Table1~DB1_Table59	DB1_Table1	2020/01/01:00:00	2020/01/01:12:00	20

【 図 16 】

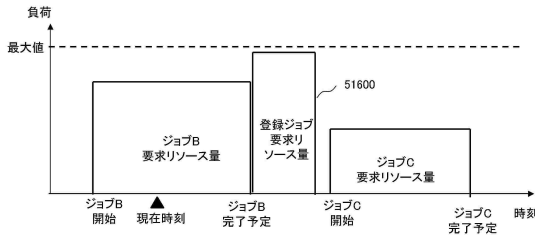


10

20

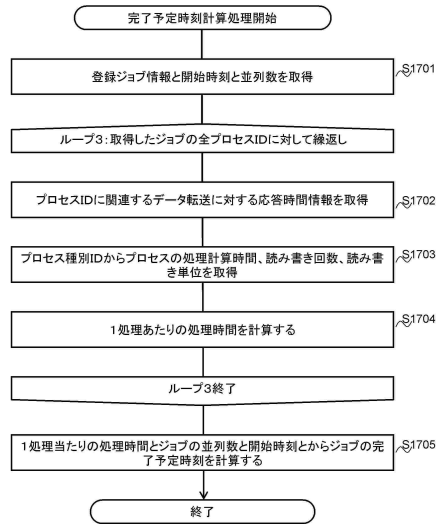
【 図 17 】

図 17



【 図 18 】

図 18



30

40

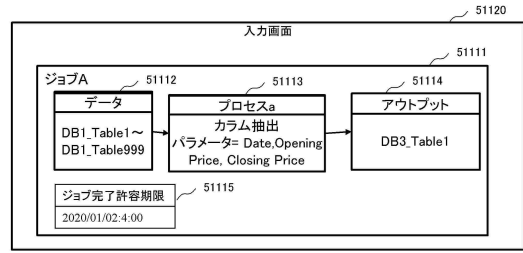
50

【 図 1 9 】

出力画面 51220

51211	51212	51213	51214
ジョブ	推奨並列数	推奨開始時刻	完了予定時刻
ジョブA	20	2020/01/02: 0:00	2020/01/02: 1:00

【 図 2 0 】



10

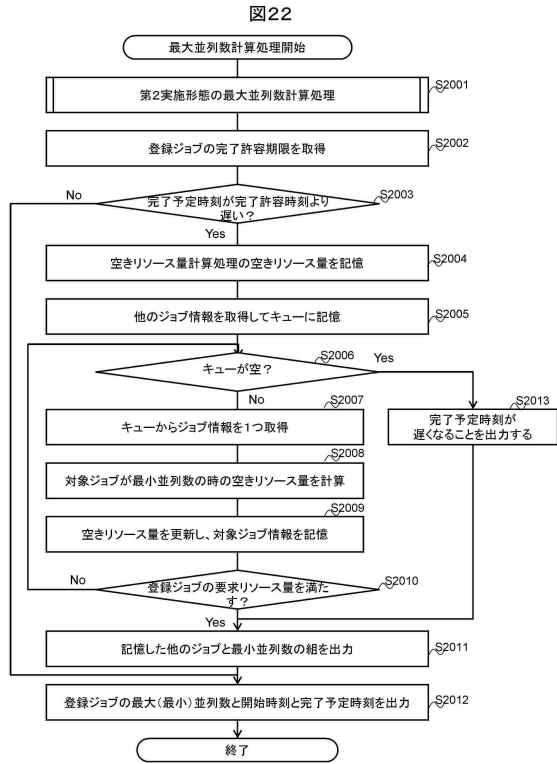
【 図 2 1 】

図 21
登録ジョブ情報テーブル860

811	812	813	814	815	816	851	852	853
ジョブID	プロセスID	プロセス種別ID	パラメータ	データソース	アウトプット	開始時刻	完了予定時刻	並列数
ジョブA	プロセスA	Table_to_Table_Extract_Column	Date opening price, closing price	DB1_Table1~DB1_Table999	DB3_Table1	2020/01/01 0:00	2020/01/01 12:00	20

861	862	863	864
要求リソース量	許容期限	最小並列数	最小要求リソース量
ネットワークIFxxx: 受信転送速度: 2Gbps ネットワークxxx: 上り転送速度: 2Gbps ネットワークxxx: 送信転送速度: 250MB/s ネットワークIFxxx: 受信転送速度: 2Gbps ネットワークxxx: 下り転送速度: 2Gbps ネットワークxxx: 送信転送速度: 250MB/s I/Oポートxxx: 受信転送速度: 125MB/s ポートAxxx: 受信転送速度: 125MB/s	2020/01/02 4:00	10	ネットワークIFxxx: 受信転送速度: 1Gbps ネットワークxxx: 上り転送速度: 1Gbps ネットワークxxx: 送信転送速度: 250MB/s ネットワークIFxxx: 受信転送速度: 1Gbps ネットワークxxx: 下り転送速度: 1Gbps ネットワークxxx: 送信転送速度: 250MB/s I/Oポートxxx: 受信転送速度: 125MB/s ポートAxxx: 受信転送速度: 125MB/s

【 図 2 2 】



20

30

40

50

フロントページの続き

審査官 久々宇 篤志

- (56)参考文献 特開 2 0 0 9 - 2 9 4 8 1 0 (J P , A)
特開 2 0 1 7 - 1 8 2 4 3 5 (J P , A)
国際公開第 2 0 1 5 / 1 9 8 4 4 0 (W O , A 1)
- (58)調査した分野 (Int.Cl. , D B 名)
- G 0 6 F 1 1 / 3 4
G 0 6 F 1 1 / 3 0
G 0 6 F 9 / 5 0