

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6110517号

(P6110517)

(45) 発行日 平成29年4月5日 (2017.4.5)

(24) 登録日 平成29年3月17日 (2017.3.17)

(51) Int.Cl.

F I

G 0 6 F 12/00 (2006.01)

G 0 6 F 12/00 5 1 0 B

G 0 6 F 12/00 5 0 1 B

G 0 6 F 12/00 5 1 1 A

請求項の数 19 (全 20 頁)

(21) 出願番号 特願2015-561907 (P2015-561907)
 (86) (22) 出願日 平成25年8月19日 (2013.8.19)
 (65) 公表番号 特表2016-515250 (P2016-515250A)
 (43) 公表日 平成28年5月26日 (2016.5.26)
 (86) 国際出願番号 PCT/CN2013/081757
 (87) 国際公開番号 W02015/024160
 (87) 国際公開日 平成27年2月26日 (2015.2.26)
 審査請求日 平成27年9月15日 (2015.9.15)

(73) 特許権者 503433420
 華為技術有限公司
 HUAWEI TECHNOLOGIES
 CO., LTD.
 中華人民共和国 518129 広東省深
 ▲チェン▼市龍崗区坂田 華為總部▲ベン
 ▼公樓
 Huawei Administrati
 on Building, Bantia
 n, Longgang Distric
 t, Shenzhen, Guangd
 ong 518129, P. R. Ch
 ina

(74) 代理人 100107766
 弁理士 伊東 忠重

最終頁に続く

(54) 【発明の名称】 データオブジェクト処理方法及び装置

(57) 【特許請求の範囲】

【請求項 1】

データオブジェクトを1つ以上のブロックに分割するステップと、
 各ブロックのサンプル圧縮率を計算し、同じサンプル圧縮率特性を有する連続ブロック
 を1つのデータセグメントに集約し、前記データセグメントのそれぞれの前記サンプル圧
 縮率を取得するステップと、

前記データセグメントのそれぞれの長さが属する長さの範囲と、前記データセグメント
 のそれぞれの前記サンプル圧縮率が属する圧縮率範囲とに従って、前記データセグメント
 をデータチャンクに分割するための期待長を選択するステップであり、前記データセグメ
 ントのそれぞれの前記サンプル圧縮率は、前記圧縮率範囲の1つに属し、前記データセグ
 メントのそれぞれの前記長さは、前記長さの範囲の1つに属するステップと

を有するデータオブジェクト処理方法。

【請求項 2】

データオブジェクトを1つ以上のブロックに分割するステップは、
 前記データオブジェクトを1つ以上の固定長ブロックに分割するステップである、請求
 項 1 に記載の方法。

【請求項 3】

隣接するデータセグメントについて、前のデータセグメントの終わりのデータチャンク
 と、次のデータセグメントの開始のデータチャンクとを接合データチャンクに接合するス
 テップを更に有する、請求項 1 又は 2 に記載の方法。

10

20

【請求項 4】

前記接合データチャンクを複数のデータチャンクに分割するステップを更に有し、
分割に使用される前記期待長は、前記前のデータセグメントに対応する期待長以下であり、分割に使用される前記期待長は、前記次のデータセグメントに対応する期待長以下である、請求項 3 に記載の方法。

【請求項 5】

データオブジェクトを 1 つ以上のブロックに分割するステップは、
コンテンツ・デファインド・チャンキング方法を使用して前記データオブジェクトをスキャンすることにより、異なる期待長を有する複数のグループの候補チャンク境界を計算し、前記複数のグループの候補チャンク境界のうち 1 つを使用することにより、前記データオブジェクトを 1 つ以上の可変長ブロックに分割するステップである、請求項 1 に記載の方法。

10

【請求項 6】

前記データセグメントをデータチャンクに分割するための期待長を選択するステップは、
、
複数のグループの候補チャンク境界から、前記データセグメントをデータチャンクに分割するための前記選択された期待長に対応する候補チャンク境界のグループを選択するステップである、請求項 1 又は 5 に記載の方法。

【請求項 7】

同じサンプル圧縮率特性を有する連続ブロックを 1 つのデータセグメントに集約するステップは、

20

前記サンプル圧縮率が同じ圧縮率範囲に属する隣接する連続ブロックを 1 つのデータセグメントに集約するステップ、又は

前記サンプル圧縮率の値の間の差が指定された閾値未満である隣接する連続ブロックを 1 つのデータセグメントに集約するステップである、請求項 1 に記載の方法。

【請求項 8】

前記方法の後に、

前記データチャンクのそれぞれのフィンガープリントを計算し、前記フィンガープリントを使用することにより、前記データチャンクのそれぞれが記憶デバイスに記憶されているか否かを決定し、前記記憶デバイスに記憶されていない前記データチャンクを、記憶されていない前記データチャンクに対応する前記フィンガープリント及び前記サンプル圧縮率と共に前記記憶デバイスに送信するステップと、

30

前記記憶デバイスにより、受信したデータチャンクの前記フィンガープリントを記憶し、前記受信したデータチャンクの前記サンプル圧縮率が圧縮率閾値に従うか否かを決定し、前記サンプル圧縮率が前記圧縮率閾値に従う前記データチャンクを圧縮して記憶するステップと

を更に有する、請求項 1 又は 3 に記載の方法。

【請求項 9】

前記データチャンクが取得されたセグメントの前記サンプル圧縮率は、前記データチャンクの前記サンプル圧縮率として使用される、請求項 8 に記載の方法。

40

【請求項 10】

前記データセグメントの前記サンプル圧縮率は、

前記データセグメントを形成する前記ブロックの前記サンプル圧縮率の算術平均を計算すること、又は

前記データセグメントを形成する前記ブロックの前記サンプル圧縮率をフラグ値として使用し、前記ブロックの長さを重みとして使用することにより、前記ブロックの前記サンプル圧縮率の加重平均を計算することである、請求項 1 に記載の方法。

【請求項 11】

データオブジェクトを 1 つ以上のブロックに分割するように構成されたブロック分割モジュールと、

50

各ブロックのサンプル圧縮率を計算し、同じサンプル圧縮率特性を有する連続ブロックを1つのデータセグメントに集約し、前記データセグメントのそれぞれの前記サンプル圧縮率を取得するように構成されたデータセグメント生成モジュールと、

前記データセグメントのそれぞれの長さが属する長さの範囲と、前記データセグメントのそれぞれの前記サンプル圧縮率が属する圧縮率範囲とに従って、前記データセグメントをデータチャンクに分割するための期待長を選択するように構成されたデータチャンク生成モジュールであり、前記データセグメントのそれぞれの前記サンプル圧縮率は、前記圧縮率範囲の1つに属し、前記データセグメントのそれぞれの長さは、前記長さの範囲の1つに属するデータチャンク生成モジュールと

を有するデータオブジェクト処理装置。

10

【請求項12】

前記ブロック分割モジュールは、前記データオブジェクトを1つ以上の固定長ブロックに分割するように構成される、請求項11に記載の装置。

【請求項13】

前記データチャンク生成モジュールは、隣接するデータセグメントについて、前のデータセグメントの終わりのデータチャンクと、次のデータセグメントの開始のデータチャンクとを接合データチャンクに接合するように更に構成される、請求項11又は12に記載の装置。

【請求項14】

前記データチャンク生成モジュールは、前記接合データチャンクを複数のデータチャンクに分割するように更に構成され、分割に使用される前記期待長は、前記前のデータセグメントに対応する期待長以下であり、分割に使用される前記期待長は、前記次のデータセグメントに対応する期待長以下である、請求項13に記載の装置。

20

【請求項15】

前記ブロック分割モジュールは、コンテンツ・デファインド・チャンキング方法を使用して前記データオブジェクトをスキャンすることにより、異なる期待長を有する複数のグループの候補チャンク境界を計算し、前記複数のグループの候補チャンク境界のうち1つを使用することにより、前記データオブジェクトを1つ以上の可変長ブロックに分割するように更に構成される、請求項11に記載の装置。

【請求項16】

前記データセグメントをデータチャンクに分割するための期待長を選択することは、複数のグループの候補チャンク境界から、前記データセグメントをデータチャンクに分割するための前記選択された期待長に対応する候補チャンク境界のグループを選択することである、請求項11又は15に記載の装置。

30

【請求項17】

前記データセグメント生成モジュールは、

各ブロックの前記サンプル圧縮率を計算し、前記サンプル圧縮率が同じ圧縮率範囲に属する隣接する連続ブロックを1つのデータセグメントに集約し、前記データセグメントのそれぞれの前記サンプル圧縮率を取得するように、或いは

各ブロックの前記サンプル圧縮率を計算し、前記サンプル圧縮率の値の間の差が指定された閾値未満である隣接する連続ブロックを1つのデータセグメントに集約し、前記データセグメントのそれぞれの前記サンプル圧縮率を取得するように構成される、請求項11に記載の装置。

40

【請求項18】

前記データチャンクのそれぞれのフィンガープリントを計算し、前記フィンガープリントを使用することにより、前記データチャンクのそれぞれが記憶デバイスに記憶されているか否かを決定し、前記記憶デバイスに記憶されていない前記データチャンクを、記憶されていない前記データチャンクに対応する前記フィンガープリント及び前記サンプル圧縮率と共に前記記憶デバイスに送信するように構成されたデータチャンク送信モジュールを更に有し、

50

前記記憶デバイスは、受信したデータチャンクの前記フィンガープリントを記憶し、前記受信したデータチャンクの前記サンプル圧縮率が圧縮率閾値に従うか否かを決定し、前記サンプル圧縮率が前記圧縮率閾値に従う前記データチャンクを圧縮して記憶するように構成される、請求項 11 に記載の装置。

【請求項 19】

請求項 1 ないし 10 のうちいずれか 1 項に記載の方法をコンピュータに実行させるプログラムを記録したコンピュータ読み取り可能記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

10

本発明は、情報技術の分野に関し、特にデータオブジェクト処理方法及び装置に関する。

【背景技術】

【0002】

重複データ除去 (Duplicate Data Elimination) としても知られるデータ重複排除 (Data Deduplication) は、データ記憶及び／又はデータ送信の効率を改善するために、データセット又はデータストリーム内の重複コンテンツを識別及び除去する処理であり、略して重複排除又は重複除去と呼ばれる。一般的に、重複排除技術では、データセット又はデータストリームは、一連のデータユニットに分割され、重複した 1 つのみのデータユニットを保持する。これにより、データ記憶処理における空間コスト又はデータ送信処理における帯域幅の消費を低減する。

20

【0003】

重複コンテンツが容易に識別されることができ、データオブジェクトをデータユニットに分割する方法は、解決される必要がある主要な問題である。データオブジェクトがデータユニットに分割された後に、データチャンクのハッシュ値 $h(\cdot)$ は、フィンガープリントとして計算され得る。同じフィンガープリントを有するデータユニットは、重複データとして規定される。従来技術では、重複排除のために一般的に使用されるデータユニットは、ファイル、固定長ブロック (Block)、コンテンツにより規定された可変長チャンク (Chunk) 等を含む。コンテンツ・デファインド・チャンキング (Content Defined Chunking、CDC) 方法は、データセット又はデータストリームを可変長チャンク系列に分割するため、データをスキャンして予め設定された特性に従うバイト列を識別し、チャンク境界としてのバイト列の位置に印を付けるためにスライディングウィンドウを採用する。この方法では、チャンク境界は、データのコンテンツ特性に基づいて選択され、同様のファイル又はデータストリームにより共有されるデータユニットを更に正確に識別することができる。従って、この方法は、様々なデータ重複排除の解決策に広く適用されている。研究によれば、コンテンツ・デファインド・チャンキング方法がデータセット又はデータストリームを分割するために採用される場合、更に細かいチャンキング粒度は、重複データの更に高い識別確率及び更に高い重複排除結果を意味する。しかし、更に細かいチャンキング粒度は、所与のデータセットから分割される更に多くの数のチャンクを意味し、これにより、インデックス化のオーバーヘッド及び重複データを検索する複雑性を増加させる。その結果、データ重複排除の時間効率が低減される。

30

40

【0004】

期待長は、チャンキング粒度を制御するためのコンテンツ・デファインド・チャンキング (Content Defined Chunking、CDC) 方法の主要なパラメータである。一般的に、CDC 方法は、特定のデータオブジェクトについて可変長チャンク系列を出力する。様々なチャンクの長さは、統計的に正規分布に従い、期待長は、正規分布の平均値を調整するために使用される。一般的に、正規分布の平均値は、平均チャンク長により表される。確率変数が正規分布のもとでの最高の確率の平均値に割り当てられるため、平均チャンク長はまた、ピーク長とも呼ばれ、理想的な状況における期待長に等しくなり得る。例えば、CDC 方法では、スライディングウィンドウ内のデータのフィンガープリント $f(w\text{-bytes})$ は、リア

50

ルタイムで計算される。f(w-bytes)の特定のビットが予め設定された値に合致する場合、スライディングウィンドウの位置は、チャンク境界として選択される。データコンテンツの更新は、ハッシュフィンガープリントのランダムな変化を生じ得るため、f(w-bytes)&0xFFF=0が合致条件として設定され、ただし、&は2進数フィールドにおけるビットのAND演算であり、0xFFFFは4095の16進数表現である場合、1つのフィンガープリントの合致は、理論上でf(w-bytes)の4096個のランダムな変化で生じる可能性がある。すなわち、チャンク境界は、スライディングウィンドウが4KB（4096バイト）前にスライドする毎に見つけられ得る。理想的な状況のもとでのチャンク長は、CDC方法における期待チャンク長（Expected Chunk Length）であり、略して期待長と呼ばれる。

【0005】

10

重複排除の空間効率を維持しつつ、できるだけチャンクの数进行低減するために、従来技術は、コンテンツ・デファインド二方式チャンキング方法を提供する。コンテンツ・デファインド二方式チャンキング方法の中心の概念は、2つの異なる期待長を有する可変長チャンキングモードを採用することである。ファイルをデータチャンクに分割する場合、重複排除記憶システムに問い合わせることにより候補チャンクの重複を決定し、重複データと非重複データとの間の遷移の領域ではスモールチャンクモードを採用し、非遷移領域ではラージチャンクモードを採用する。

【0006】

しかし、この技術は独立して動作することができず、どのようにデータオブジェクトをチャンキングするかを決定する場合、チャンク計算デバイスは、重複排除記憶デバイスに存在するデータチャンクのフィンガープリントを頻繁に問い合わせる必要がある。重複排除記憶デバイスは、データ重複排除が実行されたデータチャンクを記憶する。そして、候補チャンクの重複に従って、重複データと非重複データとの間の遷移の領域が存在するかどうかを決定し、どのチャンキングモードが最終的に採用されるかを決定する必要がある。従って、従来技術は、重複排除記憶デバイスに問い合わせ負荷の压力を引き起こす。

20

【発明の概要】

【0007】

本発明の実施例は、データオブジェクトをデータチャンクに分割することができるデータオブジェクト処理技術を提供する。

【0008】

30

第1の態様によれば、本発明の実施例は、データオブジェクト処理方法を提供し、この方法は、データオブジェクトを1つ以上のブロックに分割するステップと、各ブロックのサンプル圧縮率を計算し、同じサンプル圧縮率特性を有する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得するステップと、データセグメントのそれぞれの長さが属する長さの範囲と、データセグメントのそれぞれのサンプル圧縮率が属する圧縮率範囲とに従って、データセグメントをデータチャンクに分割するための期待長を選択するステップであり、データセグメントのそれぞれのサンプル圧縮率は、圧縮率範囲の1つに一意に属し、データセグメントのそれぞれの長さは、長さの範囲の1つに一意に属するステップとを含む。

【0009】

40

第2の態様によれば、本発明の実施例は、データオブジェクト処理装置を提供し、この装置は、データオブジェクトを1つ以上のブロックに分割するように構成されたブロック分割モジュールと、各ブロックのサンプル圧縮率を計算し、同じサンプル圧縮率特性を有する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得するように構成されたデータセグメント生成モジュールと、データセグメントのそれぞれの長さが属する長さの範囲と、データセグメントのそれぞれのサンプル圧縮率が属する圧縮率範囲とに従って、データセグメントをデータチャンクに分割するための期待長を選択するように構成されたデータチャンク生成モジュールであり、データセグメントのそれぞれのサンプル圧縮率は、圧縮率範囲の1つに一意に属し、データセグメントのそれぞれの長さは、長さの範囲の1つに一意に属するデータチャンク生成モジュール

50

ールとを含む。

【0010】

本発明の実施例の適用では、データオブジェクトはブロックに分割され、ブロックは各ブロックの圧縮率に従ってデータセグメントに集約され、データセグメントはデータチャンクに分割される。このことは、データオブジェクトをデータチャンクに分割するという結果を達成する。

【図面の簡単な説明】

【0011】

本発明の実施例又は従来技術の技術的解決策を明確に説明するために、以下に、実施例又は従来技術を説明するために必要な添付図面について簡単に紹介する。以下の説明における添付図面は、本発明の単にいくつかの実施例を示しているに過ぎず、依然として他の図面もこれらの添付図面から導かれ得る。

【図1】データオブジェクト処理方法の実施例のフローチャート

【図2】データオブジェクト処理方法の実施例のフローチャート

【図3】データオブジェクト処理方法の実施例のフローチャート

【図4】データオブジェクト処理方法の実施例のフローチャート

【図5】データオブジェクト処理方法の実施例のフローチャート

【図6】データオブジェクト処理装置の実施例の概略図

【発明を実施するための形態】

【0012】

以下に、本発明の実施例の添付図面を参照して、本発明の実施例の技術的解決策を明確且つ完全に説明する。明らかに、説明する実施例は、本発明の実施例の全てではなく、一部に過ぎない。本発明の実施例に基づいて得られる全ての他の実施例は、本発明の保護範囲内に入るものとする。

【0013】

データチャンクの期待長を低減することは、高い重複排除率を取得する助けとなるが、データチャンクの数及び対応するインデックスを増加させる可能性があり、これにより、重複データチャンクを検索する複雑性を増加させ、重複排除性能を制限する可能性がある。

【0014】

従来技術では、細粒度チャンキングは、重複コンテンツと非重複コンテンツとの間の遷移の領域に採用されるが、粗粒度チャンキングは、他の領域に採用され、これにより、二方式チャンキング方法を形成する。しかし、この方法は、チャンキング処理において候補チャンクの重複を頻繁に問い合わせることを必要とし、従って、重複排除記憶システムに問い合わせ負荷の圧力を引き起こす。更に、この方法のチャンキング結果は、複数バージョンのデータの送信順序に依存するため、この方法は不安定である。

【0015】

本発明の実施例は、圧縮性を提供し、データオブジェクトをデータチャンクに分割することができるコンテンツ・デファインド多方式データ重複排除チャンキング方法を提供する。具体的に、本発明の実施例の適用では、データオブジェクトはブロックに分割され、ブロックは各ブロックのサンプル圧縮率に従ってデータセグメントに集約され、各データセグメントをデータチャンクに分割するためにデータセグメントの長さ及びサンプル圧縮率に従って期待長が選択される。このことは、多方式の長さ分布においてデータオブジェクトをデータチャンク系列に分割するという結果を達成する。

【0016】

本発明の実施例では、データオブジェクトは、動作可能なデータのセグメント、例えば、ファイル又はデータストリームを示す。チャンキングポリシー・マッピングテーブルは、圧縮率範囲と長さの範囲と期待チャンク長との間のマッピング関係を維持管理するために使用される。データセグメントの期待チャンク長は、データセグメントの長さ及び圧縮率の増加と共に大きくなる。

【0017】

本発明の実施例で提供される方法では、データオブジェクトはデータチャンクに分割されてもよく、分割の後に取得されたデータチャンクは、データ重複排除の単位として使用されてもよい。これにより、データ損失なしに記憶空間利用率が低減されるという前提でデータオブジェクトが記憶又は送信され得る。明らかに、分割から生成されたデータチャンクは、データ重複排除以外の目的に更に使用されてもよい。本発明の実施例は、(a)データオブジェクト(Data Object)をチャンキングデバイスに入力するステップであり、データオブジェクトは、チャンキングデバイスの外又は中のメモリから生じててもよく、例えば、データオブジェクトは、重複排除動作要件を満たす限りファイル又はデータストリームでもよく、これは本発明の実施例では限定されないステップと、(b)データオブジェクトを1つ以上のブロック(Block)に分割し、サンプリング方法を使用することにより各ブロックの圧縮率を推定し、チャンキングポリシー・マッピングテーブルに問い合わせ、圧縮率と同じ圧縮率範囲に属する隣接する連続ブロックを1つのデータセグメント(Segment)に集約するステップであり、サンプル圧縮率は、データチャンクの圧縮性の尺度であり、サンプル圧縮率は、サンプリングが全体ブロックで実行された場合にブロックの圧縮率と同じであるステップと、(c)各データセグメントについてチャンキングポリシー・マッピングテーブルに問い合わせ、データセグメントに対応する圧縮率範囲及び長さの範囲に従って期待長を選択し、コンテンツ・デファインド・チャンキング方法を使用することにより、選択された期待長に従ってデータセグメントをチャンク(Chunk)系列に分割するステップと、(d)異なるデータセグメントにある隣接するチャンクを接合し、データ重複排除のための各チャンクのハッシュ値を計算するステップであり、接合ステップは任意選択のステップであり、ハッシュ値は接合なしに直接計算されてもよいステップとを含む。ステップ(b)において、データオブジェクトは、ブロック系列のサンプル圧縮率情報を使用することにより、データセグメント系列に再び分割される。複数の分割方法が存在してもよく、例えば、他の実現方式では、ステップ(b)において、圧縮率範囲に参照が行われなくてもよく、代わりにサンプル圧縮率の値の間の差が指定された閾値未満にある隣接する連続ブロックが1つのデータセグメントにされてもよい。

【0018】

<実施例1>

図1を参照すると、以下は、本発明の実施例によるデータオブジェクト処理方法を詳細に説明した具体的なステップを使用する。

【0019】

ステップ11: 重複排除されるべきデータオブジェクトを1つ以上のブロックに分割し、各ブロックのサンプル圧縮率を計算し、同じ圧縮率特性を有する隣接する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得する。各ブロックは、固定長でもよく、可変長でもよい。ブロックが可変長である場合、特定の範囲内のランダムな長さがブロックの長さとして選択されてもよい。或いは、データオブジェクトは、異なる期待長を有する候補チャンク境界の複数のグループを出力するようにスキャンされてもよく、候補チャンク境界の複数のグループのうち1つがデータオブジェクトを1つ以上の可変長ブロックに分割するために使用される。

【0020】

圧縮率は、データが圧縮可能な程度を測定するために使用され、圧縮率=圧縮データの量/元のデータの量として計算される。圧縮率を推定する方法は、サンプリングレートSに基づいて各ブロックからサンプルデータのセグメントを抽出し、データ圧縮アルゴリズム、例えば、可逆圧縮アルゴリズムLZアルゴリズム及びRLE符号化を使用することによりサンプルデータを圧縮し、圧縮率を計算し、サンプルの圧縮率をサンプリングされたブロックの圧縮率として使用するものである。更に高いサンプリングレートは、サンプルの圧縮率がサンプリングされたブロックの実際の圧縮率に更に近くなることを意味する。

【0021】

圧縮可能なデータセグメントでは、その圧縮率は一般的に1未満である。圧縮不可能な

データセグメントでは、記述フィールドのようなメタデータのオーバーヘッドの追加のため、圧縮符号化の後のデータセグメントの長さは、元のデータ長より大きくなってもよい。このことは、圧縮率が1より大きくなることを引き起こしてもよい。圧縮率のエントリは、一連の圧縮率範囲を含み、異なる圧縮率範囲の間の交わりは空であり、全ての圧縮率範囲の結合は、完全な圧縮率の値の範囲 $[0, \infty)$ を構成する。

【0022】

圧縮率特性は、ブロックを集約するパラメータとして圧縮率を使用することを示す。圧縮率又はブロックの圧縮レート演算から取得された値が予め設定された条件を満たす場合、ブロックは圧縮率特性に従う。具体的に、圧縮率特性は、圧縮率範囲、すなわち、圧縮率の範囲でもよく、また、隣接するブロックの圧縮率の値の間の差の閾値でもよい。表1を参照すると、圧縮率範囲が圧縮率特性として使用されている。データオブジェクトは、ブロック1~7の7個のブロックに分割される。各ブロックでサンプリングを実行し、これらの圧縮率を推定することにより、各ブロックのサンプル圧縮率が取得される。例えば、ブロック1のサンプル圧縮率は0.4であり、ブロック2のサンプル圧縮率は0.42である。各サンプル圧縮率は、1つの圧縮率範囲に属する。例えば、圧縮率0.4は圧縮率範囲 $[0, 0.5)$ に属し、圧縮率0.61は圧縮率範囲 $[0.5, 0.8)$ に属する。各ブロックは1つのサンプル圧縮率に対応するため、各ブロックが1つの圧縮率範囲に属するとも考えられてもよい。同じ圧縮率範囲に属するブロックは1つのデータセグメントに集約される。従って、ブロック1及びブロック2はデータセグメント1に集約されてもよく、ブロック3、ブロック4及びブロック5はデータセグメント2に集約されてもよく、ブロック6及びブロック7はデータセグメント3に集約されてもよい。

【表1】

表1

ブロック	サンプル圧縮率	圧縮率範囲	データセグメント
ブロック1	0.4	$[0, 0.5)$	データセグメント1
ブロック2	0.42		
ブロック3	0.53	$[0.5, 0.8)$	データセグメント2
ブロック4	0.61		
ブロック5	0.79		
ブロック6	0.82	$[0.8, \infty)$	データセグメント3
ブロック7	0.89		

【0023】

ステップ12：各データセグメントの長さが属する長さの範囲と、各データセグメントのサンプル圧縮率が属する圧縮率範囲とに従って、データセグメントをデータチャンクに分割するための期待長を選択する。データセグメントの長さの値の範囲は、長さの範囲の少なくとも1つに分類され、データセグメントのそれぞれの長さは、長さの範囲の1つに一意に属し、データセグメントのそれぞれのサンプル圧縮率は、サンプル圧縮率範囲の1つに一意に属する。

【0024】

異なる圧縮率範囲は交わらず、異なる長さの範囲は交わらず、圧縮率範囲及び長さの範囲のそれぞれの組み合わせは、1つの期待長に対応する。各データセグメントのサンプル圧縮率は、1つの圧縮率範囲に一意に属し、各データセグメントの長さは、1つの長さの範囲に一意に属する。データセグメントのサンプル圧縮率は、データセグメントにおいてサンプリング及び圧縮を実行することにより取得されてもよく、データセグメントを形成するブロックのサンプル圧縮率の平均値を計算することにより取得されてもよい。平均値は、算術平均でもよい。或いは、各ブロックの長さに基づいてブロックのサンプル圧縮率

の加重平均を計算することにより取得されてもよい。具体的な方法は、データセグメントを形成するブロックのサンプル圧縮率をフラグ値として使用し、ブロックの長さを重みとして使用することにより、ブロックのサンプル圧縮率の加重平均を計算することである。

【0025】

データセグメントをデータチャンクに分割する場合、データチャンクの境界が見つけれ
る必要がある。データチャンクは、隣接する境界の間にある。

【0026】

ステップ11で使用されるブロックを分割する方法が、異なる期待長を有する候補チャンク境界の複数のグループを計算し、候補チャンク境界の複数のグループのうち1つを使用することにより、データオブジェクトを1つ以上の可変長ブロックに分割することである
場合、ステップ12は、選択された期待長に従って、候補チャンク境界の複数のグループから、データセグメントをデータチャンクに分割するための同じ期待長を有するチャンク境界を選択することでもよい。すなわち、データセグメントを再びスキャンする代わりに、
ステップ11におけるスキャンにより取得された候補チャンク境界の複数のグループから対応するチャンク境界を選択する。他の方法、例えば、固定長ブロックを使用する方法が
ブロックを分割するためにステップ11で使用される場合、ステップ12は、データセグメントをデータチャンクに分割するために、期待長を選択し、データセグメントをスキャンすることにより各ブロックの境界を見つけることでもよい。異なる期待長を有する候補チャンク境界の複数のグループを計算する方法と比べて、後者の方法は、データセグメントをス
キャンすることによりデータチャンクの境界を見つける1つ以上のステップを有する。

【0027】

表2の例を参照すると、圧縮率の値の範囲は、3つの圧縮率範囲[0,0.5)、[0.5,0.8)及び[0.8,∞)に分割される。圧縮率範囲の間に交わりは存在せず、各データセグメントのサンプル圧縮率は、1つの圧縮率範囲に属する。このことはまた、各データセグメントが1つの圧縮率範囲に対応することとして理解されてもよい。更に、データセグメントの長さの値の範囲は、少なくとも1つの長さの範囲に分割される。長さの範囲の間に交わりは存在せず、従って、各データセグメントは、1つの長さの範囲に対応する。長さの範囲及び圧縮率範囲は、併せてデータセグメントの期待長を決定する。例えば、データセグメントAの長さの範囲が[0MB,10MB)であり、データセグメントAのサンプル圧縮率が圧縮率範囲[0,0.5)内にあり、長さの範囲[0MB,10MB)及び圧縮率範囲[0,0.5)により併せて決定された期待長が32KBである場合、データセグメントAの期待長は32KBである。同様に、データセグメントBの長さが属する長さの範囲と、データセグメントBのサンプル圧縮率が属する圧縮率範囲とに従って、データセグメントBの期待長が256KBであることが取得されてもよい。期待長が取得された後に、各データセグメントは、期待長に従ってデータチャンクに分割されてもよい。例えば、データセグメントAは、期待長32KBに従ってデータチャンクに分割され、データセグメントBは、期待長256KBに従ってデータチャンクに分割される。KBにおけるBはバイト(Byte)の省略であり、1KB=1024Bytesであり、1MB=1024KBである。

【0028】

期待長は理論値であるため、データチャンクの実際の長さは異なってもよい。すなわち、データチャンクは可変長である。しかし、データセグメントAから分割されたデータチャンクの平均チャンク長は、期待長32KBに近く、従って、可変長のこれらのデータチャンクの長さの値が32KBであることが最高の確率になり、32KBがピーク長でもある。本発明の実施例では、各期待長は1つのピーク長に対応し、従って、本発明の実施例は、複数のピーク長が存在するデータチャンキング及び重複排除方法を提供する。

【表 2】

表 2

データセグメント	圧縮率範囲	長さの範囲	期待長
データセグメント A	[0, 0.5)	[0 MB, 10 MB)	32 KB
データセグメント B	[0, 0.5)	[10 MB, ∞)	256 KB
データセグメント C	[0.5, 0.8)	[0, 10 MB)	64 KB
データセグメント D	[0.5, 0.8)	[10 MB, ∞)	256 KB
データセグメント E	[0.8, ∞)	[0, 100 MB)	256 KB
データセグメント F	[0.8, ∞)	[100 MB, 500 MB)	2 MB
データセグメント G	[0.8, ∞)	[500 MB, ∞)	5 MB

10

【0029】

この実施例では、期待長は、経験値を使用することにより、又は分析統計を用いて取得されてもよい。表 2 に示すように、期待長を決定するための任意選択のルールは、圧縮率範囲が同じである場合、長さの範囲の下限値が増加するほど対応する期待長が増加し、長さの範囲が同じである場合、圧縮率範囲の下限値が増加するほどマッピングテーブルの対応する期待長が増加することである。データセグメントの期待チャンク長は、データセグメントのサンプル圧縮率が属する圧縮率範囲の下限値と、データセグメントの長さが属する長さの範囲の下限値とに正相関する。或る場合には、大きいデータオブジェクト及び圧縮することが困難なデータオブジェクトの重複排除率は、チャンキング粒度にとって敏感ではなく、期待長を使用したチャンキング方法は、重複排除率を急速に悪化させることなくチャンクの数を急速に低減することができる。

20

【0030】

データセグメント内のデータチャンクの位置に基づいて、データセグメントから分割されたデータチャンクは順序になっており、順序におけるデータチャンクはデータチャンク系列を形成する。データチャンク系列は、チャンキングサブ系列と呼ばれてもよい。ステップ12はまた、分割からの異なるデータチャンクにより生成されたチャンキングサブ系列を接合し、データオブジェクトのチャンキング系列を形成することを含んでもよい。チャンキング系列におけるチャンクの順序は、データオブジェクトにおけるチャンクデータの順序と同じである。

30

【0031】

ステップ13：隣接するデータセグメントについて、前のデータセグメントの終わりのデータチャンクと、次のデータセグメントの開始のデータチャンクとをデータチャンクに接合する。接合の後に形成されたデータチャンクは、接合データチャンクと呼ばれてもよい。ステップ13は任意選択のステップである。例えば、ステップ11のデータセグメントの境界が固定長ブロックの境界である場合、ステップ13が使用されてもよく、接合は、固定長ブロックの境界がチャンクの境界として使用されることを回避することができる。このことは、更に良い重複排除効果を達成し得る。

40

【0032】

更に、2つのデータセグメントの間の遷移の領域で生成された接合データチャンクでは、接合データチャンクの2つの側における隣接するデータチャンクは、それぞれ異なる期待長に対応する。任意選択で、2つの期待長の間の更に小さい値は、接合データチャンクを更に細かい粒度のデータチャンクに分割するために採用されてもよい。これにより、2つのデータセグメントの間の遷移の領域において重複コンテンツが更に良好に識別される。このことは、データチャンクの総数のわずかな増加を引き起こしつつ、重複排除効果

50

を改善する。他の実施例では、2つの期待長の間の更に小さい値未満の期待長も、接合データチャンクを更に細かい粒度のデータチャンクに分割するために採用されてもよい。

【0033】

ステップ12がチャンキングサブ系列を接合するステップを含む場合、ステップ13は、チャンキングサブ系列が接合される前に実行されてもよく、また、チャンキングサブ系列が接合された後に実行されてもよい。

【0034】

ステップ14：データチャンクのそれぞれのフィンガープリントを計算し、フィンガープリントを使用することにより、データチャンクのそれぞれが記憶デバイスに記憶されているか否かを決定し、記憶デバイスに記憶されていないデータチャンクを、記憶されてい

10

【0035】

フィンガープリントは、データチャンクを一意に識別するために使用され、データチャンク及びデータチャンクに対応するフィンガープリントは、1対1の対応関係を有する。フィンガープリント計算方法は、データチャンクのハッシュ (Hash) 値をフィンガープリントとして計算することである。

【0036】

ステップ15：記憶デバイスは、受信したデータチャンク及びデータチャンクのフィンガープリントを記憶する。記憶中に、記憶デバイスは、受信したデータチャンクのサンプル圧縮率が圧縮率閾値に従うか否かを決定してもよい。そして、記憶空間を節約するために、圧縮率閾値に従うデータチャンクを圧縮して記憶し、圧縮率閾値に従わないデータを、そのデータを圧縮することなく直接記憶する。例えば、圧縮率閾値が0.7以下である場合、サンプル圧縮率が0.7以下であるデータチャンクは、圧縮された後に記憶されてもよく、サンプル圧縮率が0.7より大きいデータチャンクは、圧縮されることなく直接記憶される。

20

【0037】

ステップ11において1つより多くのセグメンテーション方式が使用されてもよい点に留意すべきである。例えば、他の実現方式では、ステップ11で使用されるセグメンテーションポリシーは、サンプル圧縮率の値の間の差が指定された閾値未満である隣接する連続ブロックを1つのデータセグメントに集約することとして変更されてもよい。換言すると、サンプル圧縮率の値の間の差が指定された閾値未満であることが、圧縮率特性である。依然として表1が例として使用される。サンプル圧縮率の値の間の差が0.1未満であることが閾値として使用されることを仮定する。この場合、0.42-0.4<0.1であるため、ブロック1及びブロック2は1つのデータセグメントに配置される。0.53-0.42>0.1であるため、ブロック3及びブロック2は1つのデータセグメントに配置されない。類推して、ブロック1及びブロック2は第1のデータセグメントに配置され、ブロック3及びブロック4は第2のデータセグメントに配置され、ブロック5、ブロック6及びブロック7は第3のデータセグメントに配置される。

30

【0038】

40

<実施例2>

図2に示すように、実施例2について詳細に説明する。この実施例によるデータオブジェクト処理方法は以下のステップを含む。

【0039】

21：チャンキングポリシー・マッピングテーブルをロードする。表1及び表2を参照すると、チャンキングポリシー・マッピングテーブルは、どの圧縮率が同じ圧縮率特性を有するかを記録する。更に、チャンキングポリシー・マッピングテーブルは、データセグメントの長さが属する長さの範囲と、データセグメントの圧縮率が属する圧縮率範囲と、長さの範囲及び圧縮率範囲により併せて決定される期待長とを更に記録する。期待長は、データセグメントをデータチャンクに分割するために使用される。このステップはまた、

50

チャンキングポリシー・マッピングテーブルが使用される必要がある前にロードされる限り、後に実行されてもよい。

【0040】

22：処理される必要のあるデータオブジェクトを、この方法を実現する装置に入力する。換言すると、処理される必要のあるデータオブジェクトを取得する。データオブジェクトを取得する源は、この方法を実現する装置でもよく、この方法を実現する装置に接続された外部の装置でもよい。

【0041】

23：固定長を使用することによりデータオブジェクトを分割する。具体的な実現では、このステップにおいて、データオブジェクトは、各固定長ブロックの境界を取得するためにスキャンされる必要がある。2つの隣接する境界の間のデータはブロックである。

10

【0042】

24：各ブロックのデータにおいてサンプリング及び圧縮を実行し、サンプリング及び圧縮により取得された圧縮率を全体ブロックのデータサンプル圧縮率として使用する。

【0043】

25：チャンキングポリシー・マッピングテーブルに従って同じ圧縮率特性を有する連続ブロックをデータセグメントに集約する。

【0044】

26：チャンキングポリシー・マッピングテーブルに従って各データセグメントを指定された期待長のデータチャンクに分割する。具体的な実現では、このステップにおいて、各データセグメントは、データチャンクの境界を見つけるためにスキャンされる必要があり、データセグメントは、一連のデータチャンクを形成するために境界を使用することにより分割される。

20

【0045】

この実施例では、ステップ26が実行された後に、データオブジェクトの処理が完了し、データオブジェクトをデータチャンクに分割する効果が達成される。後のステップ27、28及び29は、データオブジェクト処理方法に対する更なる拡張である。

【0046】

27：各データセグメントにより生成されたチャンキングサブ系列を接合する。各データセグメントを分割することにより形成されたデータチャンクは、データセグメントにおけるデータチャンクの位置と同じ順序になっており、順序におけるデータチャンクは、チャンキングサブ系列と呼ばれてもよい。このステップにおいて、接合方式は、いずれか2つの隣接するデータセグメントについて、データセグメントの間の境界を除去し、複数のチャンキングサブ系列を結合し、全体データオブジェクトのデータチャンク系列を形成するように、前のデータセグメントの終わりのデータチャンクと、次のデータセグメントの開始のデータチャンクとをデータチャンクに接合することである。任意選択で、このステップにおいて接合されたデータチャンクは、更に細かい粒度のデータチャンクに分割されてもよい。このことは、重複排除効果を改善する。他の接合方法は、全体データオブジェクトのデータチャンク系列を形成するために、各データチャンクを変更せずに保持したまま、次のデータセグメントの開始のデータチャンクが前のデータセグメントの終わりのデータチャンクによって先行される、データオブジェクトにおけるデータセグメントの順序に従って、各データセグメントのチャンキングサブ系列をソートすることである。

30

40

【0047】

28：チャンクのハッシュ値をフィンガープリントとして計算する。

【0048】

29：チャンク及び各チャンクのフィンガープリントを出力し、任意選択で、各チャンクのサンプル圧縮率も出力されてもよい。チャンクのサンプル圧縮率は、チャンクが属するセグメントのサンプル圧縮率を使用することによる計算を通じて取得されてもよく、1つの方式は、セグメントのサンプル圧縮率をそのセグメントのチャンクのサンプル圧縮率として直接使用することである。

50

【0049】

1つより多くのデータオブジェクトが存在する場合、全てのデータオブジェクトが処理されるまでステップ22~29を繰り返す。

【0050】

前述のステップを実現する具体的な方法は、w-bytesのスライディングウィンドウを使用することによりデータオブジェクトをスキャンし、それぞれ1バイトの前進スライディングにおいてラピッド・ハッシュ・アルゴリズムを採用することにより、ウィンドウ内のデータのフィンガープリント $f(w\text{-bytes})$ を再計算し、現在のデータセグメントの期待長がEである場合、式 $\text{Match}(f(w\text{-bytes}), E) = D$ が有効であるか否かに基づいて、現在のフィンガープリントがチャンク境界フィルタ基準を満たすか否かを決定することである。ただし、整数 $D \in [0, E)$ は、予め規定された固有値であり、関数 $\text{Match}(\cdot)$ は、 $f(w\text{-bytes})$ を範囲 $[0, E)$ にマッピングするために使用される。ハッシュ関数 $f(\cdot)$ はランダムであるため、フィルタ基準は、ピーク長がEである一連のチャンクを出力することができる。この処理が新たなデータセグメントに到達した場合、新たなデータセグメントの期待長Eが変化する。従って、新たなピーク長のチャンキングサブ系列が出力される。2つのデータセグメントの隣接する領域でチャンキングを実行する場合、データチャンクの前者の境界は、前のデータセグメントの期待長により決定され、後者の境界は、次のデータセグメントの期待長により決定される。従って、データチャンクの前者及び後者の境界は、それぞれ2つのセグメント内にあり、このことは、処理がセグメント境界をチャンク境界として使用しないように、隣接するセグメントの間のチャンキングサブ系列が自動的に接合されることと等価である。

【0051】

理解を容易にするために、実施例2は、図3に示すフローチャートを更に提供する。ステップの処理は、連続して、3a, データオブジェクト処理方法が実行される必要のあるデータオブジェクトを入力するステップであり、データオブジェクトは、データオブジェクト処理がデータオブジェクトにおいて実行される前にバッファに一時的に記憶されてもよいステップ、3b, データオブジェクトにおいてブロック分割、サンプリング及び圧縮率推定を実行するステップであり、L、M及びHはそれぞれ3つの異なる圧縮率範囲を表すステップ、3c, 同じ圧縮率特性を有する連続ブロックをデータセグメントに集約するステップ、3d, 各データセグメントの圧縮率及び長さの特性に従って期待長を選択し、チャンク境界を計算し、計算されたチャンク境界に従って各データセグメントをデータチャンクに分割するステップであり、各データセグメントから分割されたデータチャンクはチャンキングサブ系列を形成するステップ、及び3e, データセグメントのチャンキングサブ系列を接合し、チャンキングフィンガープリントを計算するステップである。

【0052】

<実施例3>

実施例2では、重複排除されるべきデータオブジェクトは、複数の固定長ブロックに分割される。しかし、実施例3では、期待長を使用することにより、重複排除されるべきデータオブジェクトにおいて事前チャンキングが実行される。具体的に、データオブジェクト（ファイル／データストリーム）は、コンテンツ・デファインド・チャンキング方法を使用することによりスキャンされ、異なる期待長を有する候補チャンク境界の複数のグループが生成される。候補チャンク境界の各グループは、データオブジェクトの1つの候補チャンキング系列に対応し、チャンキング系列はまた、その後のチャンキングの解決策として理解されてもよい。事前チャンキングの解決策の1つは、データオブジェクトをブロックに分割するために使用される。サンプリング及び圧縮は、ブロックにおいて実行され、データセグメントを分割する方法が決定され、データセグメントをデータチャンクに分割する後のステップにおいて、対応する候補チャンク境界が、期待長に従ってデータチャンクを分割するために選択される。

【0053】

従って、実施例2では、データオブジェクトがブロックに分割される場合、且つ、デー

タセグメントがデータチャンクに分割される場合、データオブジェクトがスキャンされる必要がある。この実施例では、データオブジェクトは1回のみスキャンされる必要があり、これにより、システムリソースを節約し、データオブジェクトの処理効率を改善する。更に、実施例3のブロック及び又はデータチャンクの境界は候補チャンク境界に基づくため、ブロック集約により生成されたセグメント境界は重複排除に逆効果を有さない。このことは、データチャンクのセグメント境界を除去するための動作を実行する必要はないことを意味する。換言すると、隣接するデータセグメントについて、データチャンクを形成するために、前のデータセグメントの終わりのデータチャンクと次のデータセグメントの開始のデータチャンクとを接合する必要はない。

【0054】

10

更に小さい期待長は、更に細かい平均チャンキング粒度を意味し、データコンテンツの部分圧縮率の変化を認識するのに更に助けになる。更に大きい期待長は、更に粗い平均チャンキング粒度を意味する。

【0055】

図4に示すように、この実施例で提供されるデータオブジェクト処理方法は、具体的に以下のステップを含んでもよい。

【0056】

41：チャンキングポリシー・マッピングテーブルをロードする。表1及び表2を参照すると、チャンキングポリシー・マッピングテーブルは、どの圧縮率が同じ圧縮率特性に属するかを記録する。更に、チャンキングポリシー・マッピングテーブルは、データセグメントの長さが属する長さの範囲と、データセグメントの圧縮率が属する圧縮率範囲と、長さの範囲及び圧縮率範囲により併せて決定される期待長とを更に記録する。期待長は、データセグメントをデータチャンクに分割するために使用される。このステップはまた、チャンキングポリシー・マッピングテーブルが使用される必要がある前にロードされる限り、後に実行されてもよい。

20

【0057】

42：処理される必要のあるデータオブジェクトを、この方法を実現する装置に入力する。換言すると、処理される必要のあるデータオブジェクトを取得する。データオブジェクトを取得する源は、この方法を実現する装置でもよく、この方法を実現する装置に接続された外部の装置でもよい。

30

【0058】

43：スキャン処理において、候補チャンク境界の複数のグループを出力する。候補チャンク境界の各グループ及び期待長は1対1の対応関係を有する。これらの期待長は、後のステップにおけるデータセグメントに対応する期待長を含む。

【0059】

44：ステップ43における候補チャンク境界の複数のグループから候補チャンク境界の1つのグループを選択し、選択された候補境界のグループにより形成された候補チャンキング系列においてサンプリング及び圧縮を実行する。このステップは、データオブジェクトをブロックに分割する。ブロックの期待長は、ステップ43で使用される複数の期待長のうち1つである。ブロック分割が期待長に従って実行される場合、データオブジェクトは、再びスキャンされる必要はなく、期待長に対応する候補チャンク境界に従って直接ブロックに分割される。次に、各ブロックのデータにおいてサンプリング及び圧縮を実行し、サンプリング及び圧縮により取得された圧縮率を全体ブロックデータのサンプル圧縮率として使用する。

40

【0060】

45：チャンキングポリシー・マッピングテーブルに従って同じ圧縮率特性を有する連続ブロックをデータセグメントに集約する。

【0061】

46：チャンキングポリシー・マッピングテーブルに従って、各データセグメントの期待長に対応する候補チャンク境界を選択する。データチャンクの境界は、データチャンクの

50

長さ及び位置を決定するため、このステップは、データセグメントをデータチャンクに分割する。前述のように、ステップ46における候補チャンク境界はステップ43から来たものである。このため、データセグメントがデータチャンクに分割されるときに、データオブジェクトは再びスキャンされる必要はなく、これにより、システムリソースを節約する。

【0062】

この実施例では、ステップ46が実行された後に、データオブジェクトの処理が完了し、データオブジェクトをデータチャンクに分割する効果が達成される。後のステップ47、48及び49は、データオブジェクト処理方法に対する更なる拡張である。

【0063】

47：隣接するデータセグメントのチャンキングサブ系列を接合する。データオブジェクトのデータチャンキング系列を形成するために、データオブジェクトにおけるデータセグメントの順序に従って各データセグメントのサブ系列をソートする。データチャンクはまた、チャンクと呼ばれてもよい。

【0064】

48：データチャンクのハッシュ値をフィンガープリントとして計算する。

【0065】

49：チャンク及びチャンクのフィンガープリントを出力し、任意選択で、チャンクのサンプル圧縮率も出力されてもよい。チャンクのサンプル圧縮率は、チャンクが属するセグメントのサンプル圧縮率を使用することによる計算を通じて取得されてもよく、1つの方式は、セグメントのサンプル圧縮率をそのセグメントのチャンクのサンプル圧縮率として直接使用することである。

【0066】

データチャンクを記憶する場合、チャンクのサンプル圧縮率に従って、データチャンクが記憶される前に圧縮される必要があるか否かを決定する。

【0067】

データ処理が完了していない場合、すなわち、1つより多くのデータオブジェクトが存在する場合、全てのデータオブジェクトが処理されるまで、残りのオブジェクトにおいてステップ42～49が連続して実行される。

【0068】

前述のステップ43を実現する具体的な方法は、この処理において、表2に示すチャンキングポリシー・マッピングテーブルの全ての期待長 E_i 及び対応する固有値 D_i をパラメータリストに構成し、異なるパラメータ $\langle E_i, D_i \rangle$ を連続して使用することにより、スライディングウィンドウにより出力された各フィンガープリント $f(w\text{-bytes})$ が合致条件 $\text{Match}(f(w\text{-bytes}), E_i) = D_i$ を満たすか否かを決定し、フィンガープリントが合致条件を満たす場合、現在のウィンドウ E_i に対応する候補チャンク境界が選択される。フィンガープリント合致効率は、パラメータ $\langle E_i, D_i \rangle$ を最適化することにより改善され得る。例えば、 $\text{Match}(f(w\text{-bytes}), E_i) = f(w\text{-bytes}) \bmod E_i$ が規定され、 $E_0 = 2^{12}B = 4\text{KB}$ 、 $E_1 = 2^{15}B = 32\text{KB}$ 、 $D_0 = D_1 = 0$ が選択されたときに、 $f(w\text{-bytes}) \bmod E_0 \neq D_0$ である場合、明らかに $f(w\text{-bytes}) \bmod E_1 \neq D_1$ が生じる。すなわち、1つのフィンガープリントが期待長4KBに対応するフィルタ基準を満たさない場合、フィンガープリントが期待長32KBに対応するフィルタ基準を満たすか否かを検査し続ける必要はない。

【0069】

理解を容易にするために、実施例3は、図5に示すフローチャートを更に提供する。図5のステップは、5a, データオブジェクト処理方法が実行される必要のあるデータオブジェクトを出力するステップであり、データオブジェクトは、データオブジェクト処理がデータオブジェクトにおいて実行される前にバッファに一時的に記憶されてもよいステップ、5b, 異なる期待長を使用することにより、候補チャンク境界の複数のグループを決定するステップ、5c, 5bにおいて決定された候補チャンク境界の複数のグループの中で、データオブジェクトを分割して分割後に取得されたチャンクにおいてサンプリング及び圧縮率推定を実行するために、候補チャンキング系列のグループを選択するステップ、5d, 同じ

圧縮率特性を有する連続候補ブロックをデータセグメントに集約するステップ、5e,各データセグメントの圧縮率及び長さの特性に従って期待長及び対応するチャンク境界を選択するステップであり、対応するチャンク境界に従って各データセグメントから分割されたデータチャンクはチャンキングサブ系列を形成するステップ、及び5f,データセグメントのチャンキングサブ系列を接合し、チャンキングフィンガープリントを計算するステップである。

【0070】

<実施例4>

図6を参照すると、この実施例は、実施例1、実施例2及び実施例3で提供される方法が適用されるデータオブジェクト処理装置6を説明する。データオブジェクト処理装置6は、ブロック分割モジュール61と、データセグメント生成モジュール62と、データチャンク生成モジュール63とを含む。

【0071】

データオブジェクト処理装置6において、ブロック分割モジュール61は、データオブジェクトを1つ以上のブロックに分割するように構成される。データセグメント生成モジュール62は、各ブロックのサンプル圧縮率を計算し、同じサンプル圧縮率特性を有する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得するように構成される。データチャンク生成モジュール63は、データセグメントのそれぞれの長さが属する長さの範囲と、データセグメントのそれぞれのサンプル圧縮率が属する圧縮率範囲とに従って、データセグメントをデータチャンクに分割するための期待長を選択するように構成され、データセグメントのそれぞれのサンプル圧縮率は、圧縮率範囲の1つに一意に属し、データセグメントのそれぞれの長さは、長さの範囲の1つに一意に属する。

【0072】

具体的に、ブロック分割モジュール61は、データオブジェクトを1つ以上の固定長ブロックに分割するように構成されてもよく、また、異なる期待長を有する候補チャンク境界の複数のグループを計算し、候補チャンク境界の複数のグループのうち1つを使用することにより、データオブジェクトを1つ以上の可変長ブロックに分割するように構成されてもよい。

【0073】

データセグメント生成モジュール62は、各ブロックのサンプル圧縮率を計算し、サンプル圧縮率が同じ圧縮率範囲に属する隣接する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得するように具体的に構成される。

【0074】

データセグメント生成モジュール62は、各ブロックのサンプル圧縮率を計算し、サンプル圧縮率の値の間の差が指定された閾値未満である隣接する連続ブロックを1つのデータセグメントに集約し、データセグメントのそれぞれのサンプル圧縮率を取得するように具体的に更に構成される。

【0075】

データオブジェクト処理装置6の様々な可能な実現解決策では、データチャンク生成モジュール63の機能である、データセグメントをデータチャンクに分割するための期待長を選択することは、具体的に、選択された期待長に従って、ブロック分割モジュールにより計算された候補チャンク境界の複数のグループから、データセグメントをデータチャンクに分割するための同じ期待長を有するチャンク境界を選択することでもよい。

【0076】

データチャンク生成モジュール63は、隣接するデータセグメントについて、前のデータセグメントの終わりのデータチャンクと、次のデータセグメントの開始のデータチャンクとを接合データチャンクに接合するように更に構成されてもよい。更に、データチャンク生成モジュール63は、接合データチャンクを複数のデータチャンクに分割するように更に

構成されてもよい。分割に使用される期待長は、前のデータセグメントに対応する期待長以下であり、分割に使用される期待長は、次のデータセグメントに対応する期待長以下である。

【0077】

更に、データオブジェクト処理装置6は、データチャンク送信モジュール64を更に含んでもよい。データチャンク送信モジュール64は、データチャンクのそれぞれのフィンガープリントを計算し、フィンガープリントを使用することにより、データチャンクのそれぞれが記憶デバイスに記憶されているか否かを決定し、記憶デバイスに記憶されていないデータチャンクを、記憶されていないデータチャンクに対応するフィンガープリント及びサンプル圧縮率と共に記憶デバイスに送信するように構成される。記憶デバイスは、受信したデータチャンクのフィンガープリントを記憶し、受信したデータチャンクのサンプル圧縮率が圧縮率閾値に従うか否かを決定し、圧縮率閾値に従うデータチャンクを圧縮して記憶する。

10

【0078】

データオブジェクト処理装置6はまた、CPUとメモリとにより形成されるデバイスとして考えられてもよい。メモリは、プログラムを記憶し、CPUは、メモリのプログラムに従って実施例1、実施例2及び実施例3で提供される方法を実行する。データオブジェクト処理装置6は、インタフェースを更に含んでもよい。インタフェースは、記憶デバイスに接続するために使用される。例えば、インタフェースの機能は、CPUにより処理された後に生成されたデータチャンクを記憶デバイスに送信することでもよい。

20

【0079】

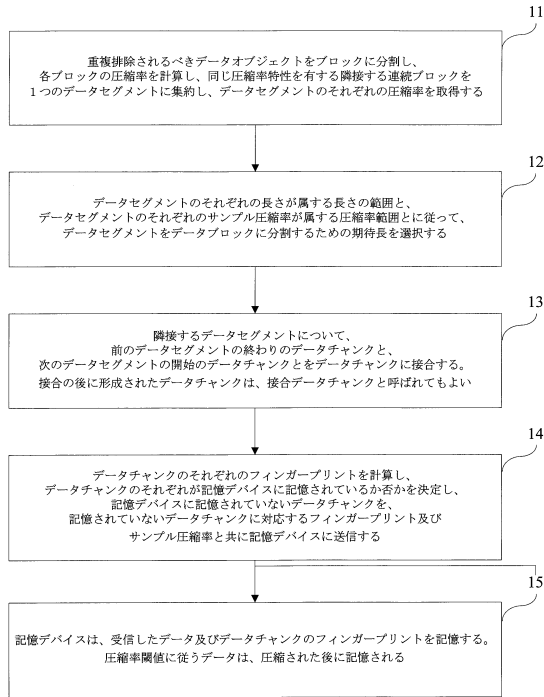
実施例の前述の説明に基づいて、当業者は、本発明が必要な一般的なハードウェアに加えてソフトウェアにより実現されてもよく、ハードウェアのみにより実現されてもよいことを明確に理解し得る。ほとんどの場合には、前者が好ましい実現方式である。このような理解に基づいて、従来技術に対して本質的に又は部分的に寄与する本発明の技術的解決策は、ソフトウェアプロダクトの形式で実現されてもよい。コンピュータソフトウェアプロダクトは、コンピュータのフロッピーディスク、ハードディスク又は光ディスクのような読み取り可能記憶媒体に記憶され、コンピュータデバイス（パーソナルコンピュータ、サーバ、ネットワークデバイス等でもよい）に対して本発明の実施例に記載の方法を実行するように命令する複数の命令を含む。

30

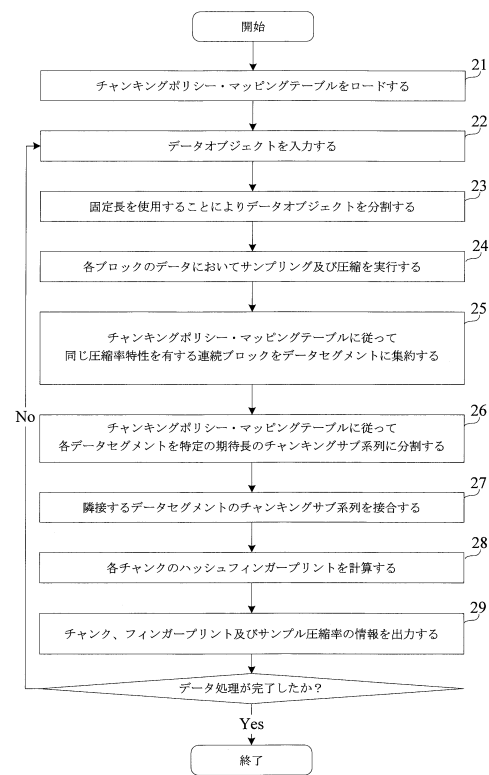
【0080】

前述の具体的な実現方式は、本発明の目的、技術的解決策及び有利な効果を詳細に記載しているが、本発明の保護範囲を限定することを意図するものではない。本発明に開示された技術的範囲内で当業者により容易に考えられる如何なる変更又は置換も、本発明の保護範囲内に入るものとする。従って、本発明の保護範囲は、特許請求の範囲の保護範囲に従うものとする。

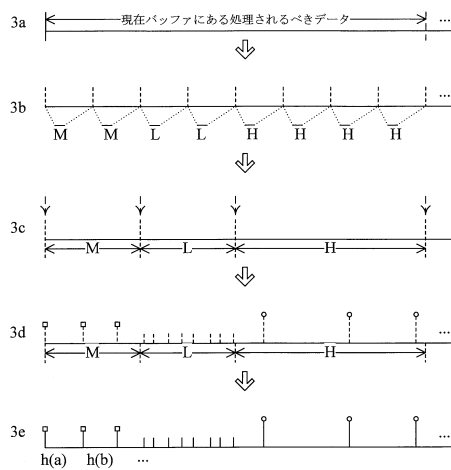
【図 1】



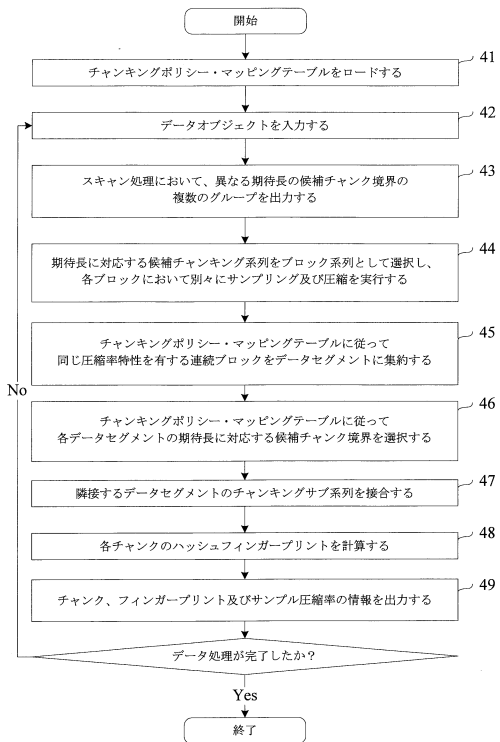
【図 2】



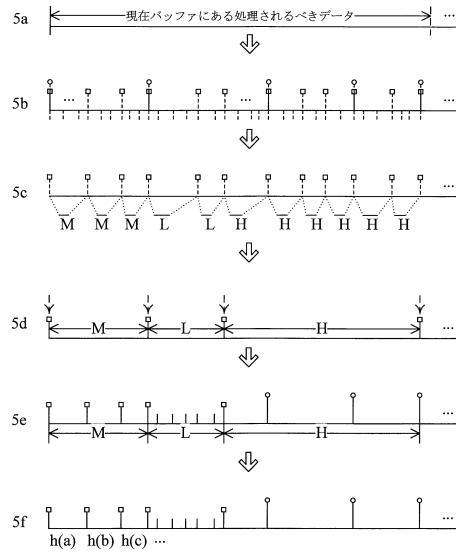
【図 3】



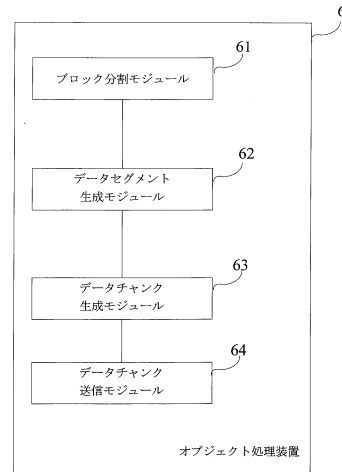
【図 4】



【図 5】



【図 6】



フロントページの続き

(74)代理人 100070150

弁理士 伊東 忠彦

(74)代理人 100091214

弁理士 大貫 進介

(72)発明者 魏 建生

中国518129▲広▼▲東▼省深▲チェン▼市▲龍▼▲崗▼区坂田▲華▼▲為▼▲総▼部▲辦▼
公樓

(72)発明者 朱 俊▲華▼

中国518129▲広▼▲東▼省深▲チェン▼市▲龍▼▲崗▼区坂田▲華▼▲為▼▲総▼部▲辦▼
公樓

審査官 原 忠

(56)参考文献 特表2013-514558 (JP, A)

特表2010-515114 (JP, A)

特開2012-164130 (JP, A)

米国特許出願公開第2013/0198148 (US, A1)

(58)調査した分野(Int.Cl., DB名)

G06F 12/00