

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5066099号
(P5066099)

(45) 発行日 平成24年11月7日(2012.11.7)

(24) 登録日 平成24年8月17日(2012.8.17)

(51) Int.Cl. F I
G06F 12/00 (2006.01)
 G06F 12/00 531R
 G06F 12/00 510B
 G06F 12/00 545A

請求項の数 69 (全 22 頁)

(21) 出願番号	特願2008-547706 (P2008-547706)	(73) 特許権者	506394175
(86) (22) 出願日	平成18年12月6日 (2006.12.6)		ソニー オンライン エンタテインメント
(65) 公表番号	特表2009-521756 (P2009-521756A)		エルエルシー
(43) 公表日	平成21年6月4日 (2009.6.4)		アメリカ合衆国 92121 カリフォル
(86) 国際出願番号	PCT/US2006/062239		ニア州, サン ディエゴ, ターマン コー
(87) 国際公開番号	W02007/073547		ト 8928
(87) 国際公開日	平成19年6月28日 (2007.6.28)	(74) 代理人	100082005
審査請求日	平成21年11月11日 (2009.11.11)		弁理士 熊倉 禎男
(31) 優先権主張番号	11/317,496	(74) 代理人	100067013
(32) 優先日	平成17年12月22日 (2005.12.22)		弁理士 大塚 文昭
(33) 優先権主張国	米国 (US)	(74) 代理人	100109070
			弁理士 須田 洋之
		(74) 代理人	100109335
			弁理士 上杉 浩

最終頁に続く

(54) 【発明の名称】 階層式セグメント化巡回冗長検査による遠隔地でのファイルの修復

(57) 【特許請求の範囲】

【請求項1】

ネットワークを介して存在するリモートオリジナルファイルに基づいてローカル破壊ファイルを修復することを、コンピュータによって実行する方法であって、

ローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成し、ローカル修復ファイルを作成するステップと、

前記ローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいて前記リモート修復ファイルの前記少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索し、前記ローカル破壊ファイルの少なくとも1つの破壊部分を特定するステップと、

前記ローカル破壊ファイルの前記少なくとも1つの破壊部分を前記リモートオリジナルファイルの前記少なくとも1つの部分に置換するステップと

を含むコンピュータによって実行する方法。

【請求項2】

前記リモートオリジナルファイルの検査コードの一連のセクションを再帰的に生成し、前記リモート修復ファイルを作成するステップ

をさらに含む、請求項1に記載のコンピュータによって実行する方法。

【請求項3】

前記リモートオリジナルファイルの前記検査コードの一連のセクションを連結し、前記リモート修復ファイルを作成するステップを

さらに含む、請求項2に記載のコンピュータによって実行する方法。

【請求項4】

前記ローカル破壊ファイルの検査コードの一連のセクションを前記再帰的に生成するステップが、

第2のバイト数が第1のバイト数より少ない場合に、前記ローカル破壊ファイルの前記第1の数のバイトに第1のハッシュを実行し、前記第2の数のバイトを有する検査コードを作成するステップと、

前記ローカル破壊ファイル全体がハッシュされ終わるまで前記第1のハッシュを繰り返して第1の複数の検査コードを作成するステップと、

を含む、請求項1に記載のコンピュータによって実行する方法。

10

【請求項5】

前記検査コードが巡回冗長検査(CRC)コードを含む、請求項4に記載のコンピュータによって実行する方法。

【請求項6】

前記第1の複数の検査コードを、それらが作成された順に第1の連結を行い、前記ローカル破壊ファイルの前記検査コードの一連のセクションの特定のセクションを形成するステップをさらに含む、請求項4に記載のコンピュータによって実行する方法。

【請求項7】

前記特定のセクションが最初に生成されたりモート検査コードセクションである、請求項6に記載のコンピュータによって実行する方法。

20

【請求項8】

前記特定のセクション内の前記第1の数のバイトに第2のハッシュを実行し、前記第2の数のバイトを有する検査コードを生成するステップと、

前記特定のセクション全体がハッシュされ終わるまで第2のハッシュを繰り返し、前記第2の複数の検査コードを作成するステップと

をさらに含む、請求項6に記載のコンピュータによって実行する方法。

【請求項9】

前記第2の複数の検査コードを、それらが作成された順に第2の連結を行い、前記ローカル破壊ファイルの前記検査コードの一連のセクションの別のセクションを形成するステップ

30

をさらに含む、請求項8に記載のコンピュータによって実行する方法。

【請求項10】

前記一連のセクションの前記別のセクションが時間シーケンスにおいて最後に生成されたりモート検査コードセクションである、請求項9に記載のコンピュータによって実行する方法。

【請求項11】

時間シーケンスにおいて前に生成されたセクションに移動し、前記前に生成されたセクションを前記特定のセクションとして指定するステップと、

前記第2のハッシュ、前記第2の連結、および前記移動を前記別のセクションの長さがしきい値より小さくなるまで繰り返すステップと

をさらに含む、請求項9に記載のコンピュータによって実行する方法。

40

【請求項12】

前記しきい値が2000バイトである、請求項11に記載のコンピュータによって実行する方法。

【請求項13】

ローカル破壊ファイルの前記検査コードの一連のセクションが、時間シーケンスにおいて最後に生成されたローカル検査コードのセクションを含む、請求項1に記載のコンピュータによって実行する方法。

【請求項14】

リモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少

50

なくとも1つの部分を前記選択的に検索するステップが、

リモートオリジナルファイルの検査コードの再帰的に生成されたセクションの、時間シーケンスにおいて最後に生成されたりリモート検査コードのセクションを検索するステップを含む、請求項13に記載のコンピュータによって実行する方法。

【請求項15】

リモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を前記選択的に検索するステップが、

前記最後に生成されたりリモート検査コードのセクションを前記最後に生成されたローカル検査コードのセクションと比較するステップを含む、請求項14に記載のコンピュータによって実行する方法。

10

【請求項16】

リモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を前記選択的に検索するステップが、

前記最後に生成されたりリモート検査コードのセクションと異なる前記最後に生成されたローカル検査コードのセクション内の検査コードにマークするステップを含む、請求項15に記載のコンピュータによって実行する方法。

【請求項17】

リモート修復ファイルおよびリモートオリジナルファイルの部分を前記選択的に検索するステップが、

ローカル破壊ファイルの検査コードのセクションおよび前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスの点で前に生成されたセクションに移動するステップと、

20

前記前に生成されたセクションを、それぞれ、前記最後に生成されたローカル検査コードおよび前記最後に生成されたりリモート検査コードとして指定するステップと、

前記検索、前記比較、前記マーク付け、前記移動、および前記指定を前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて前記前に生成されたりリモート検査コードセクションが、前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりリモート検査コードセクションになるまで繰り返すステップと

を含む、請求項16に記載のコンピュータによって実行する方法。

30

【請求項18】

リモート修復ファイルおよびリモートオリジナルファイルの部分を前記選択的に検索するステップが、

前記前のリモート検査コードセクションが前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりリモート検査コードセクションである場合に、

前記リモートオリジナルファイルの、前記マークされたコードに対応する少なくとも1つの部分をダウンロードするステップを含む、請求項17に記載のコンピュータによって実行する方法。

【請求項19】

40

前記ローカル破壊ファイルのサイズが前記リモートオリジナルファイルのサイズと一致しないときに、前記ローカル修復ファイルおよび前記リモート修復ファイルを使用し、物理的シーケンスにおいて前記ローカル破壊ファイルの始めから前記ローカル破壊ファイルの最初の破壊部分を探索するステップ

をさらに含む、請求項1に記載のコンピュータによって実行する方法。

【請求項20】

前記ローカル破壊ファイルおよび前記リモートオリジナルファイルを、前記ローカル破壊ファイルの前記探索された最初の破壊部分から再同期化するステップ

をさらに含む、請求項19に記載のコンピュータによって実行する方法。

【請求項21】

50

前記再同期化するステップが、

前記最初の破壊部分の前の最後の正しい部分と再同期化ポイントとの間の欠落データを復元し、および/または追加データを削除するステップを含む、請求項20に記載のコンピュータによって実行する方法。

【請求項22】

検査コードの現行部分は正しいと想定することにより、前記探索された最初の破壊部分から一度に1バイトずつ前方へ進むステップと、

前記ローカル破壊ファイルの前記現行部分の前記バイトを、前記ローカル破壊ファイルの前記探索された最初の破壊部分に対応する、前記リモートオリジナルファイル内のバイトと比較するステップと

をさらに含む、請求項21に記載のコンピュータによって実行する方法。

【請求項23】

新しく検出された部分が前記移動および比較のステップで検出されたとき、前記ローカル破壊ファイルの前記最後の正しい部分と前記新しく検出された部分との間の誤って追加されたバイトを破棄するステップ

をさらに含む、請求項22に記載のコンピュータによって実行する方法。

【請求項24】

前記現行部分が物理的に次の物理的なブロックより先のブロックと一致したとき、間にある少なくとも1つのブロックが削除されたことを記録するステップ

をさらに含む、請求項22に記載のコンピュータによって実行する方法。

【請求項25】

ローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成し、ローカル修復ファイルを作成する第1の検査コード生成部と、

前記ローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいて前記リモート修復ファイルの前記少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索し、前記ローカル破壊ファイルの少なくとも1つの破壊部分を特定する破壊識別子と、

前記ローカル破壊ファイルの前記少なくとも1つの破壊部分を前記リモートオリジナルファイルの前記少なくとも1つの部分に置換する置換ユニットと、

を含むシステム。

【請求項26】

前記リモートオリジナルファイルの検査コードの一連のセクションを再帰的に生成し、前記リモート修復ファイルを作成する第2の検査コード生成部

をさらに含む、請求項25に記載のシステム。

【請求項27】

前記リモートオリジナルファイルの前記検査コードの一連のセクションを連結し、前記リモート修復ファイルを作成する連結ユニット

をさらに含む、請求項26に記載のシステム。

【請求項28】

前記第1の検査コード生成部が、

前記ローカル破壊ファイルの第1の数のバイトをハッシュし、第2の数のバイトを有する検査コードを作成する第1のハッシュユニットであって、前記第2のバイト数が前記第1のバイト数より少ない、第1のハッシュユニットと、

前記ローカル破壊ファイル全体がハッシュされ終わるまで前記ハッシュを繰り返し、第1の複数の検査コードを作成する前記第1のハッシュユニットと

を含む、請求項25に記載のシステム。

【請求項29】

前記検査コードが巡回冗長検査(CRC)コードを含む、請求項28に記載のシステム。

【請求項30】

前記第1の複数の検査コードを、それらが作成された順に連結し、前記ローカル破壊フ

10

20

30

40

50

ファイルの前記検査コードの一連のセクションの特定のセクションを形成する第1の連結ユニット

をさらに含む、請求項28に記載のシステム。

【請求項31】

前記特定のセクションが第1に生成されたりモート検査コードセクションである、請求項30に記載のシステム。

【請求項32】

前記特定のセクション内の前記第1の数のバイトをハッシュし、前記第2の数のバイトを有する検査コードを生成する第2のハッシュユニットをさらに含み、

前記第2のハッシュユニットが、前記特定のセクション全体がハッシュされ終わるまで前記第1の数のバイトのハッシュを繰り返して、第2の複数の検査コードを作成する、

請求項30に記載のシステム。

【請求項33】

前記第2の複数の検査コードを、それらが作成された順に連結し、前記ローカル破壊ファイルの前記検査コードの一連のセクションの別のセクションを形成する第2の連結ユニット

をさらに含む、請求項32に記載のシステム。

【請求項34】

前記一連のセクションの前記別のセクションが時間シーケンスにおいて最後に生成されたりモート検査コードセクションである、請求項33に記載のシステム。

【請求項35】

時間シーケンスにおいて前に生成されたセクションに移動し、前記前に生成されたセクションを前記特定のセクションとして指定するためのループユニットであって、前記第2のハッシュユニット、前記第2の連結ユニット、および前記ループユニットの動作を前記別のセクションの長さがしきい値より小さくなるまで繰り返すループユニット

をさらに含む、請求項33に記載のシステム。

【請求項36】

前記しきい値が2000バイトである、請求項35に記載のシステム。

【請求項37】

ローカル破壊ファイルの前記検査コードの一連のセクションが、時間シーケンスにおいて最後に生成されたローカル検査コードのセクションを含む、請求項25に記載のシステム

【請求項38】

前記破壊識別子が、

リモートオリジナルファイルの検査コードの再帰的に生成されたセクションの、時間シーケンスにおいて最後に生成されたりモート検査コードのセクションを検索するための検索子を含む、請求項37に記載のシステム。

【請求項39】

前記破壊識別子が、

前記最後に生成されたりモート検査コードのセクションを前記最後に生成されたローカル検査コードのセクションと比較するための比較子を含む、請求項38に記載のシステム。

【請求項40】

前記破壊識別子が、

前記最後に生成されたりモート検査コードのセクションと異なる前記最後に生成されたローカル検査コードのセクション内の検査コードにマークするマークユニットを含む、請求項39に記載のシステム。

【請求項41】

前記破壊識別子が、

ローカル破壊ファイルの検査コードのセクションおよび前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスの点で前に生成されたセクションに

10

20

30

40

50

移動する第2のループユニットと、

前記前に生成されたセクションを、それぞれ、前記最後に生成されたローカル検査コードおよび前記最後に生成されたりモート検査コードとして指定するための指定ユニットであって、前記検索、前記比較、前記マーク付け、前記移動、および前記指定を前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて前記前に生成されたりモート検査コードセクションが、前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりモート検査コードセクションになるまで繰り返すための指定ユニットと

を含む、請求項40に記載のシステム。

【請求項42】

前記破壊識別子が、

前記前のリモート検査コードセクションが前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりモート検査コードセクションである場合に、前記リモートオリジナルファイルの、前記マークされたコードに対応する少なくとも1つの部分をダウンロードするダウンロードユニットを含む、請求項41に記載のシステム。

【請求項43】

前記ローカル破壊ファイルのサイズが前記リモートオリジナルファイルのサイズと一致しないときに、前記ローカル修復ファイルおよび前記リモート修復ファイルを使用し、物理的シーケンスにおいて前記ローカル破壊ファイルの始めから前記ローカル破壊ファイルの最初の破壊部分を探索する探索子

をさらに含む、請求項25に記載のシステム。

【請求項44】

前記ローカル破壊ファイルおよび前記リモートオリジナルファイルを、前記ローカル破壊ファイルの前記探索された最初の破壊部分から再同期化する再同期化ユニット

をさらに含む、請求項43に記載のシステム。

【請求項45】

前記再同期化ユニットが、

前記最初の破壊部分の前の最後の正しい部分と再同期化ポイントとの間の欠落データを復元し、および/または追加データを削除するユニットを含む、請求項44に記載のシステム。

【請求項46】

検査コードの現行部分は正しいと想定することにより、前記探索された最初の破壊部分から一度に1バイトずつ前方へ進ませる第3のループユニットと、

前記ローカル破壊ファイルの前記現行部分の前記バイトを、前記ローカル破壊ファイルの前記探索された最初の破壊部分に対応する、前記リモートオリジナルファイル内のバイトと比較する比較ユニットと

をさらに含む、請求項45に記載のシステム。

【請求項47】

新しく検出された部分が前記移動および比較で検出されたとき、前記ローカル破壊ファイルの前記最後の正しい部分と前記新しく検出された部分との間の誤って追加されたバイトを破棄する破棄ユニット

をさらに含む、請求項46に記載のシステム。

【請求項48】

前記現行部分が物理的に次の物理的なブロックより先のブロックと一致したとき、間にある少なくとも1つのブロックが削除されたことを記録する記録ユニット

をさらに含む、請求項46に記載のシステム。

【請求項49】

有形の記憶媒体に保管された、ローカル破壊ファイルを修復するコンピュータプログラムであって、コンピュータに

10

20

30

40

50

ローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成させてローカル修復ファイルを作成し、

前記ローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいて前記リモート修復ファイルの前記少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索して、前記ローカル破壊ファイルの少なくとも1つの破壊部分を特定させ、

前記ローカル破壊ファイルの前記少なくとも1つの破壊部分を前記リモートオリジナルファイルの前記少なくとも1つの部分に置換させる

実行可能な命令を含むコンピュータプログラム。

【請求項50】

コンピュータに

前記リモートオリジナルファイルの検査コードの一連のセクションを再帰的に生成し、前記リモート修復ファイルを作成させる

実行可能な命令をさらに含む、請求項49に記載のコンピュータプログラム。

【請求項51】

コンピュータに

前記リモートオリジナルファイルの前記検査コードの一連のセクションを連結し、前記リモート修復ファイルを作成させる

実行可能な命令をさらに含む、請求項50に記載のコンピュータプログラム。

【請求項52】

コンピュータに前記ローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成させる実行可能な命令が、コンピュータに

第2のバイト数が第1のバイト数より少ない場合に、前記ローカル破壊ファイルの前記第1の数のバイトに第1のハッシュを実行し、前記第2の数のバイトを有する検査コードを作成させ、

前記ローカル破壊ファイル全体がハッシュされ終わるまで前記第1のハッシュを繰り返し、第1の複数の検査コードを作成させる、

実行可能な命令を含む、請求項49に記載のコンピュータプログラム。

【請求項53】

コンピュータに

前記第1の複数の検査コードを、それらが作成された順に第1の連結を行い、前記ローカル破壊ファイルの前記検査コードの一連のセクションの特定のセクションを形成させる、

実行可能な命令をさらに含む、請求項52に記載のコンピュータプログラム。

【請求項54】

コンピュータに

前記特定のセクションの前記第1の数のバイトに第2のハッシュを実行し、前記第2の数のバイトを有する検査コードを作成させ、

前記特定のセクション全体がハッシュされ終わるまで前記第2のハッシュを繰り返し、第2の複数の検査コードを作成させる、

実行可能な命令をさらに含む、請求項53に記載のコンピュータプログラム。

【請求項55】

コンピュータに

前記第2の複数の検査コードを、それらが作成された順に第2の連結を行い、前記ローカル破壊ファイルの前記検査コードの一連のセクションの別のセクションを形成させる、

実行可能な命令をさらに含む、請求項54に記載のコンピュータプログラム。

【請求項56】

コンピュータに

時間シーケンスにおいて前に生成されたセクションに移動させ、前記前に生成されたセクションを前記特定のセクションとして指定させ、

前記第2のハッシュ、前記第2の連結、および前記移動を前記別のセクションの長さとし

10

20

30

40

50

きい値より小さくなるまで繰り返させる

実行可能な命令をさらに含む、請求項55に記載のコンピュータプログラム。

【請求項57】

ローカル破壊ファイルの前記検査コードの一連のセクションが、時間シーケンスにおいて最後に生成されたローカル検査コードのセクションを含む、請求項49に記載のコンピュータプログラム。

【請求項58】

コンピュータにリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる実行可能な命令が、コンピュータに、

リモートオリジナルファイルの検査コードの再帰的に生成されたセクションの、時間シーケンスにおいて最後に生成されたりモート検査コードのセクションを検索させる

実行可能な命令を含む、請求項57に記載のコンピュータプログラム。

【請求項59】

コンピュータにリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる実行可能な命令が、コンピュータに、

前記最後に生成されたりモート検査コードのセクションを前記最後に生成されたローカル検査コードのセクションと比較させる

実行可能な命令を含む、請求項58に記載のコンピュータプログラム。

【請求項60】

コンピュータにリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる実行可能な命令が、コンピュータに、

前記最後に生成されたりモート検査コードのセクションと異なる前記最後に生成されたローカル検査コードのセクション内の検査コードにマークさせる

実行可能な命令を含む、請求項59に記載のコンピュータプログラム。

【請求項61】

コンピュータにリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる実行可能な命令が、コンピュータに

ローカル破壊ファイルの検査コードのセクションのうち、および前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスの点で前に生成されたセクションに移動させ、

前記前に生成されたセクションを、それぞれ、前記最後に生成されたローカル検査コードおよび前記最後に生成されたりモート検査コードとして指定させ、

前記検索、前記比較、前記マーク付け、前記移動、および前記指定を前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて前記前に生成されたりモート検査コードセクションが、前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりモート検査コードセクションになるまで繰り返させる

実行可能な命令を含む、請求項60に記載のコンピュータプログラム。

【請求項62】

コンピュータにリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる実行可能な命令が、コンピュータに

前記前のリモート検査コードセクションが前記リモートオリジナルファイルの検査コードのセクションの、前記時間シーケンスにおいて最初に生成されたりモート検査コードセクションである場合に、前記リモートオリジナルファイルの、前記マークされたコードに対応する少なくとも1つの部分をダウンロードさせる

10

20

30

40

50

実行可能な命令を含む、請求項61に記載のコンピュータプログラム。

【請求項 6 3】

コンピュータに

前記ローカル破壊ファイルのサイズが前記リモートオリジナルファイルのサイズと一致しないときに、前記ローカル修復ファイルおよび前記リモート修復ファイルを使用し、物理的シーケンスにおいて前記ローカル破壊ファイルの始めから前記ローカル破壊ファイルの最初の破壊部分を探索させる、

実行可能な命令をさらに含む、請求項49に記載のコンピュータプログラム。

【請求項 6 4】

コンピュータに

前記ローカル破壊ファイルおよび前記リモートオリジナルファイルを、前記ローカル破壊ファイルの前記探索された最初の破壊部分から再同期化させる

実行可能な命令をさらに含む、請求項63に記載のコンピュータプログラム。

【請求項 6 5】

コンピュータに前記ローカル破壊ファイルおよび前記リモートオリジナルファイルを再同期化させる実行可能な命令が、コンピュータに

前記最初の破壊部分の前の最後の正しい部分と再同期化ポイントとの間の欠落データを復元させ、および/または追加データを削除させる

実行可能な命令を含む、請求項64に記載のコンピュータプログラム。

【請求項 6 6】

コンピュータに

検査コードの現行部分は正しいと想定することにより、前記探索された最初の破壊部分から一度に1バイトずつ前方へ進ませ、

前記ローカル破壊ファイルの前記現行部分の前記バイトを、前記ローカル破壊ファイルの前記探索された最初の破壊部分に対応する、前記リモートオリジナルファイル内のバイトと比較させる

実行可能な命令をさらに含む、請求項65に記載のコンピュータプログラム。

【請求項 6 7】

コンピュータに

新しく検出された部分が前記移動および比較で検出されたとき、前記ローカル破壊ファイルの前記最後の正しい部分と前記新しく検出された部分との間の誤って追加されたバイトを破棄させる

実行可能な命令をさらに含む、請求項66に記載のコンピュータプログラム。

【請求項 6 8】

コンピュータに

前記現行部分が物理的に次の物理的なブロックより先のブロックと一致したとき、間にある少なくとも1つのブロックが削除されたことを記録させる

実行可能な命令をさらに含む、請求項66に記載のコンピュータプログラム。

【請求項 6 9】

ローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成し、ローカル修復ファイルを作成する手段と、

前記ローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいて前記リモート修復ファイルの前記少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索し、前記ローカル破壊ファイルの少なくとも1つの破壊部分を特定する手段と、

前記ローカル破壊ファイルの前記少なくとも1つの破壊部分を前記リモートオリジナルファイルの前記少なくとも1つの部分に置換するための手段と

を有する装置。

【発明の詳細な説明】

【技術分野】

10

20

30

40

50

【 0 0 0 1 】

本発明は、遠隔地でのファイルの修復に関し、より具体的には、ファイルを修復するための階層式セグメント化巡回冗長検査(CRC)の使用に関する。

【背景技術】

【 0 0 0 2 】

典型的な用途では、リモートサーバがファイルの大きいインベントリを保管しており、そのファイルのうちの1つ以上にローカルクライアントが関心を持つことがある。クライアントがファイルのインベントリに保管された情報を要求し、および/またはその情報に料金を支払うと、リモートサーバは、ローカルクライアントが関心を持つファイルを伝送することができる。したがって、サーバおよびクライアントは、関心のあるファイルの同一のコピーを保持する。しかし、ファイルのクライアントコピーは、ハードウェアおよび/またはソフトウェアの誤作動、ユーザエラー、または伝送エラーの結果、破壊されることがある。

10

【 0 0 0 3 】

破壊されたファイルを訂正する1つの信頼できる方法は、ファイル全体を置換することである。多くの場合、ファイルは非常に大きい、例えば、100MB以上になることがある。このため、インターネットなどの既存の伝送チャンネルを介して置換ファイルを転送するプロセスは、高価で時間がかかってしまう。

【 0 0 0 4 】

パッチなどの他の処置とともに、破壊されたファイルの訂正を扱うことができるRsyncのような従来技術がある。

20

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 5 】

しかし、これらの技術のコストは、コストに複雑なクライアントサーバプロトコルおよびそれらを実施するために必要なサーバの稼働と維持、大量の双方向ネットワークトラフィック、セキュリティの問題などを含む可能性があるため、非常に高価になることがある。本発明の実施態様は、階層式セグメント化検査コードを使用して遠隔地でファイルを修復するための方法、装置、およびプログラムを提供する。

【課題を解決するための手段】

30

【 0 0 0 6 】

ある1つの実施態様では、遠隔地でファイルを修復する方法は、ローカル修復ファイルを作成するためにローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成するステップと、ローカル破壊ファイルの少なくとも1つの破壊部分を特定するためにローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいてリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索するステップと、ローカル破壊ファイルのその少なくとも1つの破壊部分をリモートオリジナルファイルのその少なくとも1つの部分に置換するステップを含む。

【 0 0 0 7 】

40

別の実施態様では、遠隔地でファイルを修復するシステムは、ローカル修復ファイルを作成するためにローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成するための第1の検査コード生成プログラムと、ローカル破壊ファイルの少なくとも1つの破壊部分を特定するためにローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいてリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索するための破壊識別子と、ローカル破壊ファイルのその少なくとも1つの破壊部分をリモートオリジナルファイルのその少なくとも1つの部分に置換するための置換ユニットを含む。

【 0 0 0 8 】

別の実施態様では、有形の記憶媒体に保管された、ローカル破壊ファイルを修復するた

50

めのコンピュータプログラムが開示されている。このプログラムはコンピュータに、ローカル修復ファイルを作成するためにローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成させ、ローカル破壊ファイルの少なくとも1つの破壊部分を特定するためにローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいてリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させ、ローカル破壊ファイルのその少なくとも1つの破壊部分をリモートオリジナルファイルのその少なくとも1つの部分に置換させる実行可能な命令を含む。

【0009】

さらに別の実施態様では、遠隔地でファイルを修復するための装置は、ローカル修復ファイルを作成するためにローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成する手段と、ローカル破壊ファイルの少なくとも1つの破壊部分を特定するためにローカル修復ファイルの少なくとも1つの部分とリモート修復ファイルの少なくとも1つの部分との違いに基づいてリモート修復ファイルの少なくとも1つの部分およびリモートオリジナルファイルの少なくとも1つの部分を選択的に検索させる手段と、ローカル破壊ファイルのその少なくとも1つの破壊部分をリモートオリジナルファイルのその少なくとも1つの部分に置換する手段を含む。

【発明を実施するための最良の形態】

【0010】

本発明の実施態様は、階層式セグメント化検査コードを使用して遠隔地でファイルを修復するための方法、装置、およびプログラムを提供する。

【0011】

本発明のある1つの実施態様では、リモートファイル修復技術は、ローカルファイルの比較的迅速な修復を行い、一方では、従来の技術より大幅に少ない量のデータが転送されればよい。さらに、転送は単純な既存のデータ転送プロトコルを使用することができて、いずれかの特殊な通信プロトコルまたはインタラクティブソフトウェアサーバを必要としない。

【0012】

図1Aに示されているように、クライアント110が、リモートサーバ100などのリモート記憶ユニットによって保持されているファイルのインベントリに保管されている情報を要求し、および/またはその情報に料金を支払う場合、関心のあるファイル102は、伝送チャンネル120を用いてクライアント110に伝送される。したがって、クライアント110は、リモートサーバ100に保管されている関心のあるファイル102と同一のファイルのローカルコピー112を保持する。しかし、ファイルのローカルコピー112は、ハードウェアおよび/またはソフトウェアの誤作動を含むエラー/誤作動、ユーザエラー、ディスクエラー、読取りエラー、ウィルス感染、意図的な変更、伝送エラー、および/または他の関連のエラー/誤作動の結果として破壊されることがある。クライアント110は、定期的な検査、実行に関する問題、サーバからのプロンプト(更新など)、実行時の検査、またはウィルス除去後の検査によってファイル破壊を検出できる。

【0013】

ローカル破壊ファイルは、図1で114として示されている。しかし、ローカル破壊ファイル114は、ローカルファイル112と実質的に同じ物理スペースを使用する。

【0014】

ローカル破壊ファイル114を修復するために、サーバ100およびクライアントの両方が修復ファイル104、116を計算し、ローカル自己修復をサポートする。ローカル破壊ファイル114の修復方法を検討するために、クライアント110はリモートサーバ100にファイル114の破壊部分(1つまたは複数)を特定するために必要な情報のみを要求する。サーバ100は、要求された情報を修復ファイル104から検索し、それをクライアント110に伝送し、クライアント110は、受信した情報を使用してファイル114のどの部分(1つまたは複数)が破壊されているかを判定する。クライアント110が破壊部分(1つまたは複数)を判定したら、クライ

10

20

30

40

50

アントはその部分(1つまたは複数)をサーバ100に要求する。サーバ100は、オリジナルファイル102から置換部分(1つまたは複数)を検索し、その部分(1つまたは複数)をクライアント110に送信して破壊ファイル114を正しい状態に修復または復元する。

【0015】

上述のプロセスでは、比較的少ない数のバイトが伝送チャンネル120上を転送され、貴重な帯域幅が温存(確保)され、低または中帯域幅接続のユーザにとっては必要な待機期間が短くなる。ある1つの実施態様では、サーバ100は、関心のあるファイル102をクライアント110に送信する前に修復ファイル104を計算することができる。他の実施態様では、サーバ100は、クライアント110から修復ファイル104の部分(1つまたは複数)の伝送の要求を受信した後を含めて、任意の時に修復ファイル104を計算することができる。

10

【0016】

図1Bは、本発明のある1つの実施態様によるリモートファイル修復プロセスのフローチャート130を示している。ボックス140で、ローカル破壊ファイル114の検査コードの一連のセクションが再帰的に生成されてローカル修復ファイル116が作成される。次いで、リモートオリジナルファイル102の検査コードの一連のセクションが再帰的に生成されて、ボックス142でリモート修復ファイル104が作成される。

【0017】

ローカルファイル114のサイズが144でリモートファイル102のサイズと比較される。ローカルファイルのサイズがリモートファイルのサイズと一致した場合、ボックス146でローカルファイル114の破壊部分(1つまたは複数)が検出され、探索される。破壊部分(1つまたは複数)の検出および探索は、修復ファイル104、116の中の検査コードの再帰的な比較および検索を伴う。ローカルファイルの破壊部分(1つまたは複数)がボックス148で検出され、置換される。ローカルファイルのサイズがリモートファイルのサイズと一致しない場合(ボックス144)、ボックス150で修復ファイル104、116が使用されて、ローカルファイル114の始めからローカルファイル114の第1の破壊部分が探索される。

20

【0018】

ボックス152で、ローカルファイルおよびリモートファイル114、102が、ローカルファイル114の探索された第1の破壊部分から再同期化される。再同期化は、最後の正しい部分と再同期化ポイントとの間の欠落データの復元および/または追加データの削除を含む。

【0019】

ローカルファイル114のサイズが154で再びリモートファイル102のサイズと比較される。ボックス154でローカルファイルのサイズがリモートファイルのサイズと一致した場合、156でローカル修復ファイル116が再構築され、プロセスはボックス146に進んで、ローカルファイル114の破壊部分(1つまたは複数)が検出され、探索される。ローカルファイルのサイズがリモートファイルのサイズと一致しない場合(ボックス154)、ボックス150で修復ファイル104、116が使用されて、現在の再同期化ポイントからローカルファイル114の第1の破壊部分が探索される。

30

【0020】

リモートオリジナルファイル102の検査コードの一連のセクションを再帰的に生成することによるリモート修復ファイル104の作成(ボックス142)が以下で詳しく説明され、図2および図3に示されている。ローカル破壊ファイル114の検査コードの一連のセクション(ボックス140)を再帰的に生成することによってローカル修復ファイル116を作成するためにも同様のプロセスが使用できる。ローカルファイルの破壊部分(1つまたは複数)の検出および探索(ボックス146)も以下で詳しく説明され、図4および図5に示されている。ローカルファイルおよびリモートファイルの再同期化(ボックス152)が以下で詳しく説明され、図7に示されている。

40

【0021】

図2は、「修復インデックスファイル」200の1つの実施態様を示している。ある1つの実施態様では、修復インデックスファイル200(a')は、図3に示されたフローチャート300に従って生成される。

50

【 0 0 2 2 】

最初に、オリジナルファイル210(a)のNバイトのブロックがボックス310でMバイトの検査コードにハッシュされる。ある1つの実施態様では、N:Mのブロック対検査コードの比率「100対1」が効率のよい比率である。1例では、ブロックは2000バイト長であり、検査コードは20バイト長である。ある実施態様では、検査コードは、巡回冗長検査(CRC)コードとして実現される。別の実施態様では、検査コードは、SHA-160と称されるセキュアな160ビット(20バイト)ハッシュとして実現される。しかし、検査コードは、他のタイプのハッシュとして実現されることもできる。

【 0 0 2 3 】

ボックス312で、ファイル210全体がハッシュされたかどうかを判定する検査が行われる。ファイル全体がまだハッシュされていないと判定された場合、プロセスはボックス310に戻って、オリジナルファイルの別のブロック(すなわち、Nバイト)を別のMバイト検査コードにハッシュする。そうではなく、ファイル全体がハッシュされたと判定された場合、被ハッシュ検査コードがボックス314で連結されて「セクション」(例えば、図2のCRCセクション₁)が形成される。ここで、修復インデックスファイル200のM*kからM*(k+1)番目のバイトが、オリジナルファイル210のN*kからN*(k+1)番目のバイトに対する被ハッシュ検査コードである。

10

【 0 0 2 4 】

ボックス310、312、314のステップがボックス316、318、320で繰り返されて、そのとき、連結された検査コード(例えば、CRCセクション₁)が新規「セクション」(例えば、CRCセクション₂)の検査コードのソース材料として処理される。したがって、ボックス314で形成されたセクションのNバイトは、ボックス316でMバイト検査コードにハッシュされる。ボックス318で、セクション全体(例えば、CRCセクション₁)がハッシュされたかどうかを判定する検査が行われる。セクション全体がまだハッシュされていないと判定された場合、プロセスはボックス316に戻って、形成されたセクションの別のNバイトをハッシュして、別のMバイト検査コードを形成する。そうではなく、セクション全体(例えば、CRCセクション₁)がハッシュされたと判定された場合、被ハッシュ検査コードがボックス320で連結されて新規「セクション」(例えば、CRCセクション₂)が生成される。ここで、新規セクション(例えば、CRCセクション₂)のM*kからM*(k+1)番目のバイトが、前のセクション(例えば、CRCセクション₁)のN*kからN*(k+1)番目のバイトに対する被ハッシュ検査コードである。

20

30

【 0 0 2 5 】

ボックス322で、新規の連結された(ボックス320で形成)セクションの長さがしきい値と比較されて、新規セクション内の生成された検査コードが、サーバから最小の回数のデータダウンロードで効率的なローカルファイル修復を行うのに十分に小さいことが確認される。新規セクションの長さがしきい値より長い場合、プロセスはボックス316、318、320へ戻り、その新規セクションをソース材料として使用して検査コードの別のセクションを生成する。ある1つの実施態様では、しきい値が2000バイトに設定されていて、2000バイト以下の検査コードのセクションがこのプロセスを終了させるようになっている。したがって、修復インデックスファイル200のセクションはしきい値が満たされるまで再帰的に生成される。最終的には、ボックス324で、すべてのセクションが作成順に連結され、修復インデックスファイル200(a')が形成される。図2に例示の実施態様では、ファイル210全体の検査コード(例えば、CRCコード)が計算されて、修復ファイル200の先頭に修復ファイル200のヘッダとして連結される。

40

図2は修復ファイルの1つの特定の構成を示しているが、検査コードの様々な一連のセクションを含む様々な構成が可能である。

【 0 0 2 6 】

図4は、ローカル破壊ファイル406の部分(1つまたは複数)を特定して修復するためにファイル400の部分(1つまたは複数)を選択的に検索することにより修復インデックスファイル400を使用するプロセスの1つの実施態様を示している。ある1つの実施態様における、

50

クライアントが修復インデックスファイル400を使用するプロセスが図5に示されたフローチャート500に詳しく示されている。図4および図5に示された実施態様では、クライアントは、ローカル破壊ファイルのセクション(1つまたは複数)430を特定して修復するために修復インデックスファイル400の選択的に検索された部分(複数)を使用する。クライアントは、破壊の最小単位をサーチしながら、破壊ファイルを概念的に連続的な、より小さいセクションに分割する。ある1つの実施態様では、リモート修復インデックスファイル400の小さい部分(1つまたは複数)は、標準HTTP 1.1プロトコルおよび<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.htm#sec14.35>で説明されている「バイト範囲検索」方法を使用して選択的に検索されることができる。

【0027】

最初に、ボックス510で、ローカルファイル406が修復を必要としているかどうかの判定が行われる。ある1つの実施態様では、この判定は、ローカルファイル406のCRC 422を計算することにより行われる。一部の実施態様では、CRCの中間および並行計算が行われることができ、後のディスク入出力の要求を減らすために将来の参照が可能なように保管されることができる。ローカルファイル406の計算済みCRC 422がサーバのオリジナルファイル402のCRC 458と一致する場合、そのローカルファイルは修復を必要としない。しかし、ローカルファイル406の計算済みCRC 422がサーバのオリジナルファイル402のCRC 458と一致しない場合、ローカルファイル406は破壊されており、修復を必要とする。フローチャート500のボックス512から526は、ローカルファイル406の破壊エリア(複数)430を隔離/特定し、サーバから置換部分(1つまたは複数)460を検索し、破壊エリア(複数)430をその検索された置換部分(1つまたは複数)で修復/訂正(例えば、置換によって)する修復プロセスを示している。

【0028】

クライアントは、ローカル破壊ファイル406の修復ファイル404をボックス512で生成する。ローカル破壊ファイル406の修復ファイル404を生成するプロセスは、図3に示されたリモート修復インデックスファイル400の生成をミラーする。

【0029】

ローカル破壊ファイル406の修復ファイル404が生成されると、リモート修復インデックスファイル400の最後に生成された検査コードセクション(セクション_k)450がボックス514で検索される。次いで、検索されたセクション450がボックス516でローカル修復ファイル404の対応するセクション424と比較される。セクション424内の、セクション450内の検査コードと異なる検査コード410がボックス518でマークされる。

【0030】

マークされた検査コード410は、リモート修復インデックスファイル400内の前に生成されたセクション(セクション_{k-1})の部分(1つまたは複数)452をボックス520で検索するために使用される。次いで、検索された部分(1つまたは複数)452がボックス522でローカル修復ファイル404内のセクション_{k-1}の対応する部分(1つまたは複数)412と比較される。部分(1つまたは複数)412内の、部分(1つまたは複数)452内の検査コードとは異なる検査コード414がボックス524でマークされる。

【0031】

ボックス526で、前に生成された検査コードセクション(すなわち、セクション_{k-1})が最初に生成された検査コードセクション(すなわち、セクション₁)でないと判定された場合、プロセスはボックス520から繰り返される。すなわち、検索520、比較522、およびマーキング524のステップは、すべての検査コードセクションの部分(1つまたは複数)の調べが終わるまで再帰的に繰り返される。したがって、マークされた検査コード414は、リモート修復インデックスファイル400内の検査コードセクション(すなわち、セクション_{k-2})の部分(1つまたは複数)452をボックス520で検索するために使用される。ボックス520で検索された部分(1つまたは複数)454がボックス522でローカル修復ファイル404内のセクション_{k-2}の対応する部分(1つまたは複数)416と比較される。

【0032】

10

20

30

40

50

前に生成された検査コードセクションが最初に生成された検査コードセクション(すなわち、セクション₁)であると判定された場合、サーバ内のオリジナルファイル402の、第1のセクション内のマークされた検査コード420に対応する部分(1つまたは複数)460がボックス528でクライアントにダウンロードされる。最後に、ボックス530で、サーバ内のオリジナルファイルのダウンロードされた部分(1つまたは複数)460によってローカルファイル406の破壊部分(1つまたは複数)430が置換される。

【0033】

再び図4を参照すると、オリジナルファイル402のローカルコピー406内のデータは、いくつかの異なる形で破壊される可能性がある。例えば、図6Aから6Eに示されているように、データビットは、「回転する(twiddle)」、「欠落する」または「追加される」ことによって、またはそれらの任意の組合せによる影響によって破壊される。

10

【0034】

図6Aに示された1つの例では、データビットは、「回転する」ことによって破壊され、ここでは、破壊ファイル406内のデータビットが、オリジナルファイル402内にあったビットのみで並べ替えられているか、またはオリジナルファイル402のデータビットになかった無関係なビットで並べ替えられている。したがって、同じ数のデータビットが追加され、かつ削除されることがあり、ファイル内のデータビットの長さは同じままである。

【0035】

図6Bに示された別の例では、データビットは「欠落する」ことによって破壊され、ここで、破壊ファイル406内のデータビットは、オリジナルファイル402内にあった欠落ビットである。したがって、破壊ファイル406内のデータビットの数は、オリジナルファイル402内のデータビットの数より少ない。

20

【0036】

図6Cに示された別の例では、データビットは「追加される」ことによって破壊され、ここで、破壊ファイル406内のデータビットは、オリジナルファイル402内になかった追加ビットを含む。したがって、破壊ファイル406内のデータビットの数は、オリジナルファイル402内のデータビットの数より多い。

【0037】

図6Dに示された別の例では、データビットは、ビットの回転と削除の組合せによって破壊されている。したがって、図6Dに示された例において、破壊ファイル406のデータビットは、(1)オリジナルファイル402内のビットのみで並べ替えられ、(2)オリジナルファイル402のデータビットになかった無関係なビットと一緒に並べ替えられ、(3)オリジナルファイル402にあったビットが欠落している。したがって、破壊ファイル406内のデータビットの数は、オリジナルファイル402内のデータビットの数より少ない。

30

【0038】

図6Eに示された別の例では、データビットは、回転と追加の組合せによって破壊されている。したがって、図6Eに示された例において、破壊ファイル406のデータビットは、(1)オリジナルファイル402内のビットのみで並べ替えられ、(2)オリジナルファイル402のデータビットになかった無関係なビットと一緒に並べ替えられ、および/または(3)オリジナルファイル402になかった追加ビットを含んでいる。したがって、破壊ファイル406内のデータビットの数は、オリジナルファイル402内のデータビットの数より多い。

40

【0039】

図7は、データビットが、「欠落する」または「追加される」ことによって、またはそれらの任意の組合せによる影響によって破壊されたとき、破壊エリア(複数)を特定/隔離するためのフローチャート700を示している。したがって、ローカル破壊ファイルのサイズおよびリモートオリジナルファイルのサイズが一致しない(すなわち、ビットが「欠落している」および/または「追加されている」)とき、ローカル破壊ファイルは、ボックス702で、「正しい」セクションの最初のチェーンの最後からサーチされる。サーチは、ボックス704で、完全なオリジナルブロックサイズのデータを使用して、一度に1バイトずつ前方へ進む。この場合、ファイルの終わりが現行ブロックにある場合を除き、現行ブロッ

50

クが「正しい」ブロックと想定して、1バイト前方の新しい位置から開始してオリジナルブロックサイズにわたって続行する。ボックス706で、想定された正しいブロックに対する検査コードが後続の検査コードと比較される。代替の実施態様では、サーチはサブブロックサイズ単位で実行できる。しかし、ブロックサイズが小さければ小さいほど、ダウンロード効率の向上は小さくなることを理解すべきである。

比較(ボックス706で行われる)の結果として新規ブロックが検出されると、最後の正しいブロックと新しく検出されたブロックとの間の数バイトは、ボックス708で、誤って「追加された」バイトとして破棄される。一方、想定された正しいブロックの検査コードが、比較(ボックス706で行われる)の結果として、次のブロックより先のブロックと一致したとき、ボックス710で、間にあるブロック(1つまたは複数)が削除されたことが記録される。したがって、いずれかの後続ブロックが検出され、「欠落している」または「追加された」ビットが適切に処理されると、ボックス712で、通常のモードで、ファイルの残り部分の破壊エリア(1つまたは複数)の特定/隔離が続行される。この場合、ビットの「回転(twiddled)」があると想定して特定/隔離が続行される。それ以外の場合に、ファイルの終りに達した後も後続のブロックが検出されないとき、または計算または時間の制約のためにテスト継続のしきい値が検出されると、破壊ファイルは、714で、破壊エリアから前方にダウンロードのマークが付けられる。

【 0 0 4 0 】

図8Aは、コンピュータシステム800およびユーザ802を示している。ユーザ802は、コンピュータシステム800を使用してリモート修復ファイルおよび/またはローカル修復ファイルを生じ、その修復ファイルを使用してローカル破壊ファイルの部分(1つまたは複数)を特定し、修復する。1つの例では、コンピュータシステムは、修復ファイル生成アプリケーションプログラム(修復ファイル生成プログラム)を保管して実行する。修復ファイル生成プログラム815は、上述のように、リモート修復ファイルおよび/またはローカル修復ファイルを生じ、その修復ファイルを使用してローカル破壊ファイルの部分(1つまたは複数)を特定し、修復するように設計されている。

【 0 0 4 1 】

図8Bは、図8Aのコンピュータシステム800の1つの実施態様のブロック図を、修復ファイル生成プログラムを含めて示している。コンピュータシステム800は、コントローラ810、メモリ820、ストレージ830、媒体デバイス840、ユーザインターフェース850、入出力(I/O)インターフェース860、およびネットワークインターフェース870を備えている。これらの構成要素は共通バス880によって相互接続されている。あるいは、コントローラを中心にした星形などの別の接続構成が使用されてもよい。

【 0 0 4 2 】

コントローラ810はプログラブルプロセッサであって、コンピュータシステム800およびその構成要素の動作を制御する。コントローラ810は、メモリ820または埋込みコントローラメモリ(図示せず)から命令をロードして、これらの命令を実行してシステムを制御する。その実行中、コントローラ810は、ソフトウェアシステムとして修復ファイル生成プログラム815を提供する。あるいは、このサービスは、コントローラ810またはコンピュータシステム800内の分離した構成要素として実現されてもよい。修復ファイル生成プログラム815は、修復ファイルを生じ、破壊ファイルならびにストレージ830から検索されたオリジナルファイルおよびリモート修復ファイルの選択された部分(複数)を使用してローカルファイルの破壊エリア(1つまたは複数)を隔離する。

【 0 0 4 3 】

メモリ820は、コンピュータシステム800の他の構成要素で使用するためのデータを一時的に保管する。ある1つの実施態様では、メモリ820はRAMとして実現される。ある1つの実施態様では、メモリ820は、長期または永続メモリ、例えば、フラッシュメモリおよび/またはROMなども含む。

【 0 0 4 4 】

ストレージ830は、コンピュータシステム800の他の構成要素で使用するためのデータ、

10

20

30

40

50

例えば、サーバから検索されたオリジナルファイルおよびリモート修復ファイルの選択された部分(複数)などを一時的または長期に保管する。ある1つの実施態様では、ストレージ830はハードディスクドライブである。ストレージ830は、修復ファイル生成プログラム815で使用するための情報、例えば、サーバから検索されたオリジナルファイルおよびリモート修復ファイルの選択された部分(複数)などを保管する。ストレージ830は、修復生成プログラムによって生成されたデータおよびデータ構造、例えば、マークされた検査コードなども将来使用できるように保管する。

媒体デバイス840は、リムーバブル媒体を受け入れ、挿入された媒体からデータを読み取り、および/または挿入された媒体にデータを書き込む。ある1つの実施態様では、媒体デバイス840は光ディスクドライブである。

【0045】

ユーザインターフェース850は、コンピュータシステム800のユーザからのユーザ入力を受け入れ、情報をユーザに表示するための構成要素を備えている。ある1つの実施態様では、ユーザインターフェース850は、キーボード、マウス、オーディオスピーカ、およびディスプレイを備えている。コントローラ810は、ユーザからの入力を使用してコンピュータシステム800の動作を調整する。

【0046】

入出力インターフェース860は、対応する入出力デバイス、例えば、外部ストレージまたは補足デバイス(例えば、プリンタまたはPDA)などに接続するための1つ以上の入出力ポートを備えている。ある1つの実施態様では、入出力インターフェース860のポートには、USBポート、PCMCIAポート、シリアルポート、および/またはパラレルポートなどのポートがある。別の実施態様では、入出力インターフェース860は、外部デバイスと無線で通信するための無線インターフェースを備えている。

【0047】

ネットワークインターフェース870は、有線および/または無線ネットワーク接続、例えば、イーサネット(登録商標)接続をサポートするRJ-45または「Wi-Fi」インターフェース(802.11)を備えている。

【0048】

コンピュータシステム800は、簡潔にするために構成要素として図8Bに具体的に示されていないが、コンピュータシステムに典型的な追加のハードウェアおよびソフトウェア(例えば、電力、冷却、オペレーティングシステム)を備えている。他の実施態様では、別の構成のコンピュータシステムが使用されてよい(例えば、異なるバスまたはストレージ構成、またはマルチプロセッサ構成)。

【0049】

図9は、ファイル修復システム900のブロック図を示している。システム900は、検査コード生成プログラム910、破壊識別子920、および置換ユニット930を有している。

【0050】

検査コード生成プログラム910は、リモートオリジナルファイルおよびローカル破壊ファイルの検査コードの一連のセクションを再帰的に生成するように構成されている。リモートオリジナルファイルの検査コードがリモート修復ファイルを作成する。破壊識別子920は、ローカル破壊ファイルの破壊部分を特定するために、ローカルファイルの一連の再帰的に生成されたセクションの検査コードの検出された破壊部分に基づいて、リモート修復ファイルおよびリモートオリジナルファイルの部分を選択的に検索するように構成されている。置換ユニット930は、ローカル破壊ファイルの破壊部分をリモートオリジナルファイルの検索された部分で置換するように構成されている。

【0051】

ファイル修復システム900は、システム900が、更新/変更されたセクション(1つまたは複数)のみを検出して、選択的に検索できるようにローカルファイルのバージョンを更新または変更するために使用されてよいことに留意されたい。例えば、リモートファイルがバージョンAからバージョンBにアップグレードされている場合で、バージョンAのローカ

10

20

30

40

50

ルコピーをバージョンBにアップグレードされることが望まれているとき、バージョンAのローカルコピーはローカル破壊ファイルと見なされ、バージョンBのリモートファイルはリモートオリジナルファイルと見なされてよい。したがって、ファイル修復システム900は、リモートオリジナルファイルの更新/変更されたセクション(1つまたは複数)のみを検出して選択的に検索し、一方では、リモートサーバからのデータダウンロードの量を最小にする。別法では、システムが置換すべき部分(1つまたは複数)のリストを提供することができ、クライアントが、更新された部分を要求または受信して置換することができる。

【0052】

本発明の様々な実施態様は、電子ハードウェア、コンピュータソフトウェア、またはこれらの技術を組み合わせることで実現される。実施態様によっては、プログラマブルプロセッサまたはコンピュータによって実行される1つ以上のコンピュータプログラムを有するものがある。概ねは、各コンピュータは1つ以上のプロセッサ、1つ以上のデータストレージ構成要素(例えば、揮発性または不揮発性メモリモジュールおよび永続的な光および磁気ストレージデバイス、例えば、ハードおよびフロッピー(登録商標)ディスクドライブ、CD-ROMドライブ、ならびに磁気テープドライブ)、1つ以上の入力デバイス(例えば、マウスおよびキーボード)、および1つ以上の出力デバイス(例えば、ディスプレイコンソールおよびプリンタ)を備えている。

【0053】

コンピュータプログラムは、通常は永続ストレージ媒体に保管され、実行時にメモリにコピーされる実行可能なコードを有する。プロセッサは、メモリからプログラム命令を規定の順序で検索することにより、そのコードを実行する。プログラムコードを実行するとき、コンピュータは、入力デバイスおよび/またはストレージデバイスからデータを受け取って、そのデータに演算を実行し、結果のデータを出力デバイスおよび/またはストレージデバイスに送る。

【0054】

本発明の様々な例示的な実施態様が説明されているが、当業者は、他の実施態様も可能であり、本発明の範囲内であることが理解できよう。例えば、上述の説明で用語「サーバ」および「クライアント」に言及するが、用語「クライアント」は「ローカル」であるアクションおよびファイルを指すことがあり、用語「サーバ」は、「リモート」であるアクションおよびファイルを指すことがある。別の例では、サーバファイルおよび/またはローカルファイル内のデータは圧縮されてよい。さらに、ビットが欠落している、または追加されている場合にリモートファイル内のブロックと一致するブロックをローカルファイル内で検出するには、ローリングチェックサムを使用することが望ましいことがある。別の例では、ファイル内の破壊の修復はパッチの適用に似ているので、場合によっては、またはすべての場合に、デルタ配布(および無期限のサーバサイド保留)を置換できる可能性がある。さらに別の例では、公式に従ってスタガ(stagger)または変更されたブロックサイズは、ファイルサイズまたは破壊の可能性に応じて異なる。したがって、本発明は、上述の実施態様に限定されない。

【図面の簡単な説明】

【0055】

【図1A】図1Aは、本発明のある1つの実施態様によるファイル修復システムのブロック図である。

【図1B】図1Bは、本発明のある1つの実施態様による、破壊ファイルを遠隔地で修復するプロセスのフローチャートである。

【図2】「修復インデックスファイル」のある1つの実施態様を示す図である。

【図3】修復インデックスファイルを生成するプロセスを示したフローチャートである。

【図4】ローカル破壊ファイルの部分(1つまたは複数)を特定して修復するためにファイルの部分(1つまたは複数)を選択的に検索することにより、修復インデックスファイルを使用するプロセスのある1つの実施態様を示す図である。

【図5】修復インデックスファイルを使用するプロセスを示したフローチャートである。

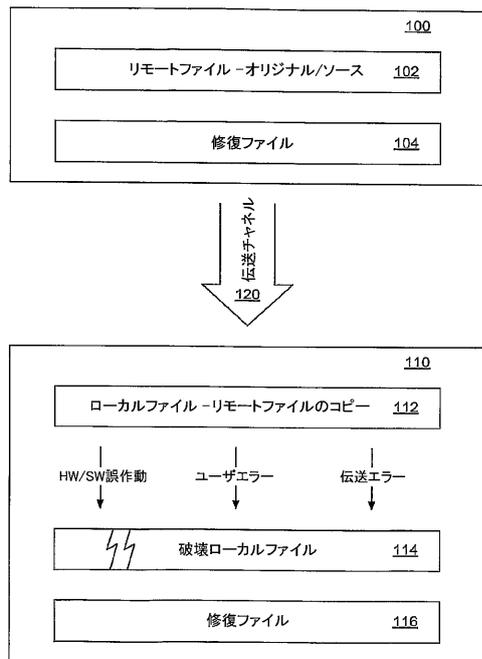
【図6】図6Aは、データビット破壊の構成を示す図である。図6Bは、データビット破壊の構成を示す図である。図6Cは、データビット破壊の構成を示す図である。図6Dは、データビット破壊の構成を示す図である。図6Eは、データビット破壊の構成を示す図である。

【図7】ファイルの破壊エリア(1つまたは複数)を特定/隔離するフローチャートである。

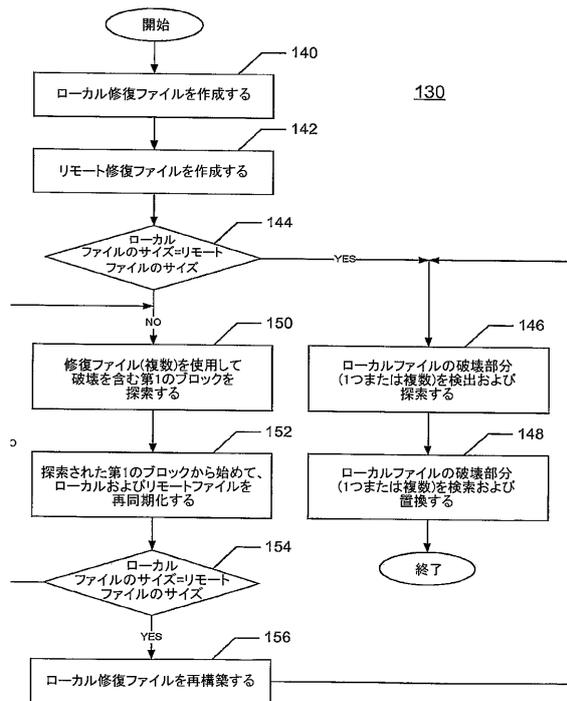
【図8】図8Aは、コンピュータシステムおよびユーザを示す図である。図8Bは、図8Aのコンピュータシステムの1つの実施態様のブロック図を、修復ファイル生成プログラムを含めて示す図である。

【図9】リモートファイル修復システムのブロック図である。

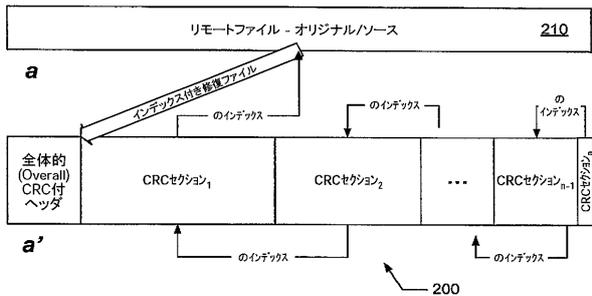
【図1A】



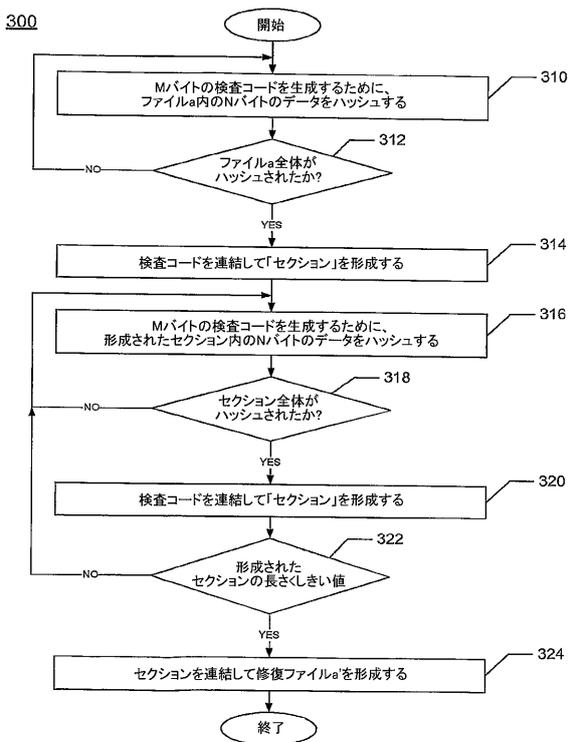
【図1B】



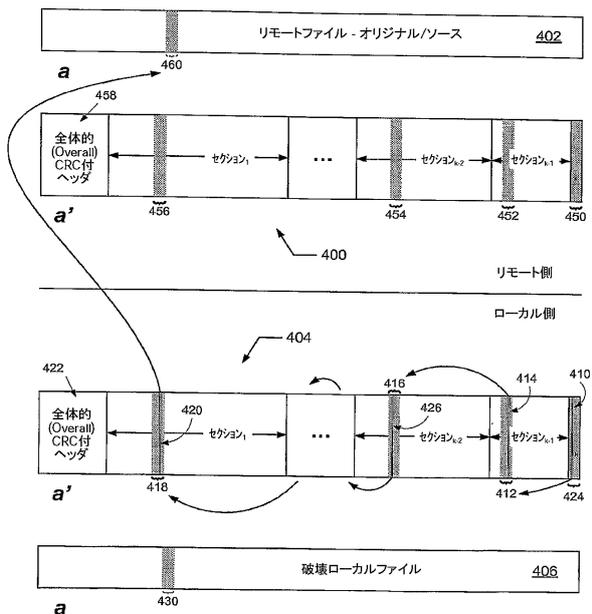
【図2】



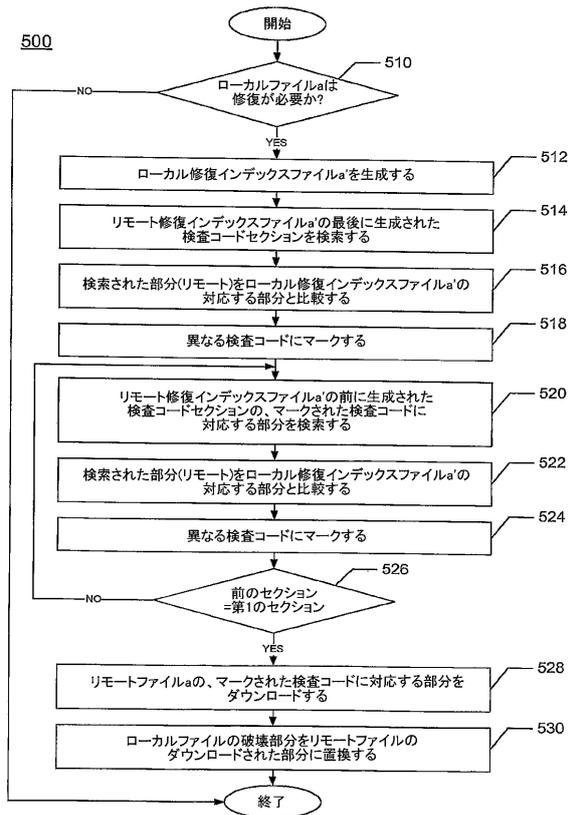
【図3】



【図4】



【図5】



【図6】

ビットの回転

A_{オリジナル} abcdefghijk
 A_{復元} bacdefghijk

6A

ビットの欠落

A_{オリジナル} abcdefghijk
 A_{復元} aghijk

6B

ビットの追加

A_{オリジナル} abcdefghijk
 A_{復元} abbbbbbcdefghijk

6C

組合せ

ビットの回転および欠落

A_{オリジナル} abcdefghijk
 A_{復元} bacdefgh
 または
 A_{オリジナル} abcdefghijk
 A_{復元} fghkij

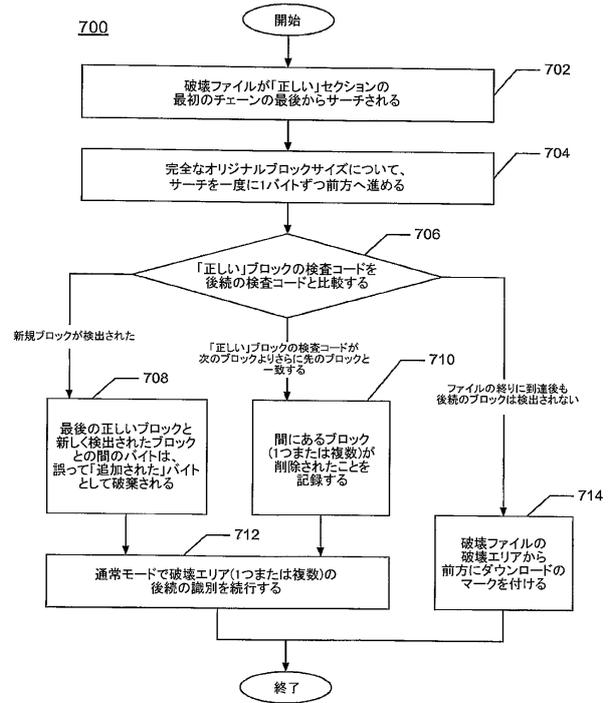
6D

ビットの回転および欠落

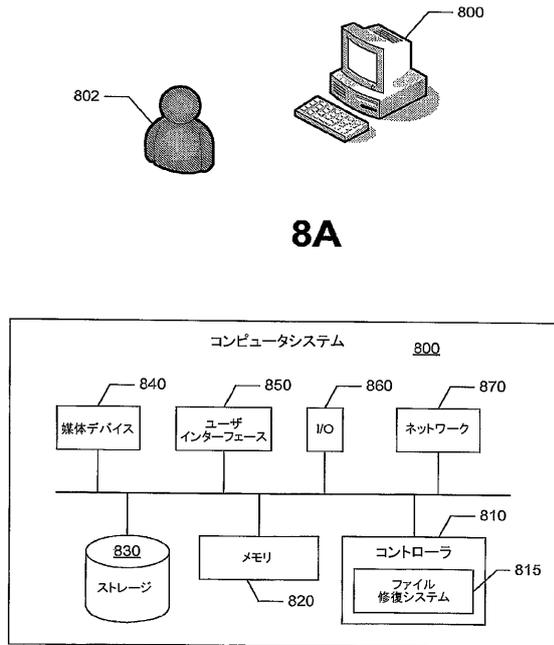
A_{オリジナル} abcdefghijk
 A_{復元} bacdefqqqqghijk
 または
 A_{オリジナル} abcdefghijk
 A_{復元} qbcdefghijkq

6E

【図7】

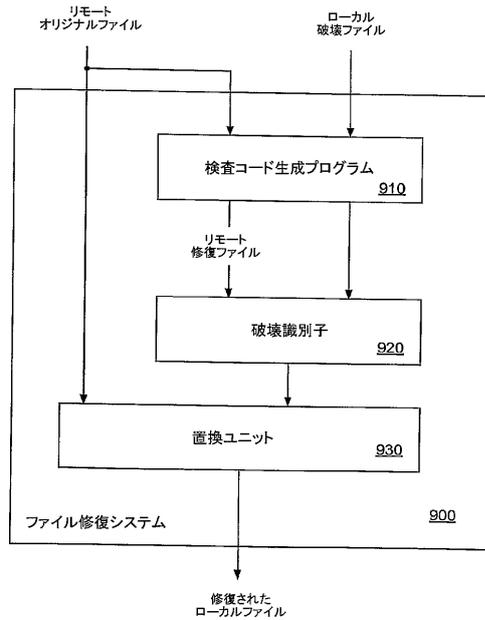


【図8】



8B

【図9】



フロントページの続き

(72)発明者 マッカーラー, パトリック
アメリカ合衆国 9 2 1 3 1 カリフォルニア州, サン ディエゴ, アイビー ヒル ドライブ
1 0 9 2 6, ナンバー 6

審査官 桜井 茂行

(56)参考文献 特開平 0 7 - 2 9 5 8 6 7 (J P , A)
米国特許第 0 5 4 4 6 8 8 8 (U S , A)
特開平 1 1 - 2 7 2 5 7 3 (J P , A)
特開 2 0 0 3 - 2 2 3 3 8 4 (J P , A)
米国特許出願公開第 2 0 0 5 / 0 2 0 4 3 5 3 (U S , A 1)
米国特許出願公開第 2 0 0 3 / 0 0 3 3 2 1 4 (U S , A 1)
米国特許第 6 0 4 9 6 1 2 (U S , A)

(58)調査した分野(Int.Cl., D B名)
G06F 12/00
G06F 11/00
JSTPlus(JDreamII)