

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3793771号

(P3793771)

(45) 発行日 平成18年7月5日(2006.7.5)

(24) 登録日 平成18年4月14日(2006.4.14)

(51) Int. Cl.

G06F 12/00 (2006.01)

F I

G06F 12/00 520E

G06F 12/00 520G

請求項の数 6 (全 31 頁)

(21) 出願番号	特願2004-225211 (P2004-225211)	(73) 特許権者	500046438
(22) 出願日	平成16年8月2日(2004.8.2)		マイクロソフト コーポレーション
(62) 分割の表示	特願平7-503643の分割		アメリカ合衆国 ワシントン州 9805
原出願日	平成6年6月30日(1994.6.30)		2-6399 レッドモンド ワン マイ
(65) 公開番号	特開2005-44370 (P2005-44370A)		クロソフト ウェイ
(43) 公開日	平成17年2月17日(2005.2.17)	(74) 代理人	100059959
審査請求日	平成16年8月2日(2004.8.2)		弁理士 中村 稔
(31) 優先権主張番号	08/085, 186	(74) 代理人	100067013
(32) 優先日	平成5年6月30日(1993.6.30)		弁理士 大塚 文昭
(33) 優先権主張国	米国 (US)	(74) 代理人	100065189
(31) 優先権主張番号	08/259, 227		弁理士 穴戸 嘉一
(32) 優先日	平成6年6月20日(1994.6.20)	(74) 代理人	100074228
(33) 優先権主張国	米国 (US)		弁理士 今城 俊夫
		(74) 代理人	100084009
			弁理士 小川 信夫

最終頁に続く

(54) 【発明の名称】 オブジェクトのリンク追跡方法及びシステム

(57) 【特許請求の範囲】

【請求項1】

ファイルシステムオブジェクトの更新によりファイルシステムオブジェクトに関連した特性を維持するためのコンピュータシステムにおける方法であって、各オブジェクトはネームを有しており、上記方法は、

第1のネームを有する第1ファイルシステムオブジェクトであって、該第1ファイルシステムオブジェクトに関する1つ以上の特性が関連された第1ファイルシステムオブジェクトを記憶し、

上記記憶された第1ファイルシステムオブジェクトのネームが第2のネームに変更されたという指示を受け取り、

上記記憶された第1ファイルシステムオブジェクトのネームが第2のネームに変更されたという指示を受け取った後の所定の時間内に、第2ファイルシステムオブジェクトのネームが第1のネームに変更されたという指示を受け取り、そして

第2ファイルシステムオブジェクトのネームが第1のネームに変更されたという指示を受け取るのに応答して、上記第1ファイルシステムオブジェクトに関連した特性を上記第2ファイルシステムオブジェクトに関連させる、
という段階を備えたことを特徴とする方法。

【請求項2】

情報を記憶する上記段階はファイルシステムオブジェクト識別子を各々記憶する請求項1に記載の方法。

【請求項 3】

ファイルシステムオブジェクトに関連したファイルシステムオブジェクト情報を更新により維持するためのコンピュータシステムにおける方法であって、

第 1 ネームを有する第 1 ファイルシステムオブジェクトを記憶し、

該第 1 ファイルシステムオブジェクトに関連して、第 1 ファイルシステムオブジェクトに関する第 1 ファイルシステムオブジェクト情報を記憶し、

第 1 ファイルシステムオブジェクトが削除されたという指示を受け取り、

第 1 ファイルシステムオブジェクトが削除されたという指示を受け取った後所定時間内に、第 1 ネームを用いて第 2 ファイルシステムオブジェクトが形成されたという指示を受け取り、そして

第 1 ネームを用いて第 2 ファイルシステムオブジェクトが形成されたという指示を受け取る段階に応答して、第 2 ファイルシステムオブジェクトに関連して第 1 ファイルシステムオブジェクトに関する情報を記憶する、
という段階を備えたことを特徴とする方法。

10

【請求項 4】

第 1 ファイルシステムオブジェクトが削除されたという指示を受け取るのに応答して、最近削除されたファイルシステムオブジェクトのネームのリストに第 1 ネームを追加し、

上記第 1 ファイルシステムオブジェクトが削除されたという指示を受け取るのに応答して、第 1 ネームをハッシュし、

上記ハッシュされた第 1 ネームを最近削除されたファイルシステムオブジェクトのハッシュされたネームのリストに追加し、

ネームを有するファイルシステムオブジェクトが形成されたという指示を受け取り、

上記形成されたファイルシステムオブジェクトのネームをハッシュし、

上記形成されたオブジェクトの上記ハッシュされたネームを、ハッシュされたネームのリストにおける最近削除されたファイルシステムオブジェクトのハッシュされたネームと比較し、

20

上記形成されたファイルシステムオブジェクトのハッシュされたネームが、ハッシュされたネームのリストにおける最近削除されたファイルシステムオブジェクトのハッシュされたネームの 1 つに一致する場合に、上記形成されたファイルシステムオブジェクトのネームをネームのリストにおける最近削除されたファイルシステムオブジェクトのネームと比較し、そして

30

上記第 2 ファイルシステムオブジェクトのネームが上記第 1 ネームと同じである場合に、上記第 1 ネームを用いて第 2 オブジェクトが形成されたという指示を与える、
という段階を更に備えた請求項 3 に記載の方法。

【請求項 5】

ファイルシステムオブジェクトの更新によりファイルシステムオブジェクトに関する情報を維持するための装置において、

第 1 のネームを有する第 1 ファイルシステムオブジェクト及び第 2 のネームを有する第 2 ファイルシステムオブジェクトを記憶するためのメモリと、

上記第 1 ファイルシステムオブジェクトに関連して、上記第 1 ファイルシステムオブジェクトに関する情報を記憶するための情報記憶サブシステムと、

40

第 1 ファイルシステムオブジェクトのネームが第 3 ネームに変更されたときを検出し、そしてその所定の時間内に、第 2 ファイルシステムオブジェクトのネームが第 1 のネームに変更されるときを検出するためのネーム転送モニタと、

第 1 ファイルシステムオブジェクトのネームが第 3 のネームに変更されそしてその所定の時間内に第 2 ファイルシステムオブジェクトのネームが第 1 のネームに変更されることを上記ネーム転送モニタが検出したときに、第 2 ファイルシステムオブジェクトに関連して第 1 ファイルシステムオブジェクトに関する情報を記憶するための情報転送サブシステムと、

を備えたことを特徴とする装置。

50

【請求項 6】

ファイルシステムオブジェクトの更新によりファイルシステムオブジェクトに関する情報を維持するための装置において、

第 1 のネームを有する第 1 ファイルシステムオブジェクトを含むファイルシステムオブジェクトを記憶するためのメモリと、

上記第 1 ファイルシステムオブジェクトに関連して、上記第 1 ファイルシステムオブジェクトに関する情報を記憶するための情報記憶サブシステムと、

第 1 ファイルシステムオブジェクトが削除されたときを検出し、そしてその所定の時間内に第 1 のネームを有する第 2 ファイルシステムオブジェクトが形成されるときを検出するための削除及び置き換えモニタと、

第 1 ファイルシステムオブジェクトが削除されそしてその所定の時間内に第 1 のネームを有する第 2 ファイルシステムオブジェクトが形成されることを上記削除及び置き換えモニタが検出したときに、第 2 ファイルシステムオブジェクトに関連して第 1 ファイルシステムオブジェクトに関する情報を記憶するための情報転送サブシステムと、
を備えたことを特徴とする装置。

10

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、1993 年 6 月 30 日に出願された米国特許出願第 08/085,186 号の一部継続出願であり、これを参考としてここに援用する。

20

【0002】

本発明は、一般に、オブジェクトに対するリンクを追跡するコンピュータ方法及びシステムに係り、より詳細には、オブジェクトに対するリンクを解明する方法及びシステムに係る。

【背景技術】**【0003】**

コンピュータシステムは、情報を含むオブジェクトを構成し、記憶し、検索しそして更新するのにしばしば使用される。ワードプロセッサ又はスプレッドシートのようなアプリケーションプログラムは、通常、これらのアクティビティを実行するのに使用される。典型的に、ユーザは、アプリケーションプログラムを用いることによってオブジェクトを構成し、キーボードのような入力装置を用いてオブジェクトの全ての内容を入力する。例えば、ユーザは、テキストの多数の序文文節、数値表及びテキストの多数の終結文節を含むレポートオブジェクトを構成すべき場合に、通常は、キーボードを用いて、序文文節、数値の表及び終結文節をタイプ打ちする。

30

【0004】

オブジェクトの内容を入力しやすくするために、ユーザは、既存のオブジェクトから、構成されているオブジェクトへ情報をコピーすることができる。このコピー方法は、ユーザが既に入力した情報の再入力を回避できるという利点を有する。例えば、ユーザがレポートオブジェクトを構成し、そして数値表を含む原簿オブジェクトが既に存在するときには、ユーザは、数値表を再タイプするのではなく、原簿オブジェクトからレポートオブジェクトへ数値表をコピーすることができる。

40

【0005】

コピー方法は、数値表が原簿オブジェクトからレポートオブジェクトへコピーされるときに、そのオブジェクトが原簿オブジェクトとの関連性を失うという欠点を有する。従って、原簿オブジェクトが変更された場合に、レポートオブジェクトの数値表は自動的に変更されない。

【0006】

関連性を失うというこの欠点は、オブジェクトリンクを使用することによって克服することができる。オブジェクトリンク（リンク）は、クライアントオブジェクトに記憶されたソースオブジェクトに対するレファレンスである。コンピュータシステムは、ソースオ

50

プロジェクトの現在内容がクライアントオブジェクトに組み込まれているかのようにリンクを処理する。ユーザがクライアントオブジェクトにアクセスするときに、コンピュータシステムは、リンクに遭遇し、そしてソースオブジェクトを探索してそれにアクセスする。リンクのソースオブジェクトを探索することを、リンクを解明すると称する。リンクが使用されるときは、ソースオブジェクトの現在バージョンがクライアントオブジェクトに組み込まれる。それ故、クライアントオブジェクトは、たとえそれがリンクの形成後に生じたとしても、ソースオブジェクトに対して更新されるという利点を有する。

【0007】

リンクの使用の一例として、ユーザは、セールス情報の数値表を含む原簿オブジェクトを、セールス情報のテキスト記述を含むレポートオブジェクトにリンクすることができる。

10

【0008】

図1は、リンクの使用を説明する図である。「report.doc」と称するレポートオブジェクト101は、「ledger.xls」と称する原簿オブジェクト103に対するリンク102を含んでいる。report.docオブジェクトが表示されたときには、ledger.xlsオブジェクトに対するリンクが解かれ、ledger.xlsオブジェクトの内容をアクセスして表示104に組み込むことができる。ここでは、ledger.xlsオブジェクトがソースオブジェクトであり、そしてreport.docオブジェクトがクライアントオブジェクトである。

【0009】

20

各リンクは、ソースオブジェクトがどこに記憶されるかを探索するのに使用される情報を含んでいる。オブジェクトは、種々の記憶装置に種々の編成で一貫して記憶することができる。例えば、ハイアラキーファイルシステムはオブジェクトをファイルとして記憶する。ハイアラキーファイルシステムは、ルートディレクトリがファイル及びサブディレクトリを含むことのできるファイルシステムである。サブディレクトリは、ファイル及び更に別のサブディレクトリを含むことができる。従って、ルートディレクトリから下降するサブディレクトリの次々のレベルがハイアラキーを形成する。パスネームはハイアラキー内の位置を記述しそして各パスネームはファイル又はサブディレクトリを参照する。例えば、パスネーム「\dos\copy.exe」は、ルートディレクトリに含まれた「dos」と称するディレクトリに含まれた「copy.exe」と称するファイルを記述する。ハイアラキーファイルシステムは、通常、リンクをパスネームとして記憶する。

30

【0010】

パスネームは、絶対的であるか又は相対的であるかのいずれかである。絶対的なパスネームは、ルートディレクトリに対してファイルを位置決めするのに必要な情報を含んでいる。一方、相対的なパスネームは、他のファイルの位置に対してファイルを位置決めするのに必要な情報を含んでいる。相対的なパスネームを含むリンクは、クライアントオブジェクトの位置に対してソースオブジェクトの位置を特定する。ソースオブジェクトがクライアントオブジェクトと同じディレクトリに位置されているときには、リンクは、キャラクタ「. \」が前に置かれたソースオブジェクトネームを含む。それ故、report.doc及びledger.xlsオブジェクトが同じディレクトリに配置された場合は、リンクのパスネームは、「. \ledger.xls」となる。絶対的なパスネームは、ルートディレクトリで始めて、ソースオブジェクトに到達するために次々に下降しなければならないサブディレクトリの順序付けされたリストである。ledger.xlsオブジェクトが「acme」と称するディレクトリにあって、これが次いで「companies」と称するディレクトリにあり、そしてこれがルートディレクトリにある場合には、ledger.xlsオブジェクトの絶対的なパスネームが「\companies\acme\ledger.xls」となる。

40

【0011】

図2は、リンクを記憶しそして解明する従来の方法を示す。クライアントオブジェクト200は、レポートオブジェクトである。これは、原簿オブジェクトに対するリンクを含

50

み、原簿オブジェクトは、次いで、ソースオブジェクトの絶対的パスネーム「\companies\acme\ledger.xls」を含む。上記のように、この絶対的なパスネームは、ファイルシステムハイアラキーにおける位置を特定する。ファイルシステムハイアラキーは、ディレクトリ220ないし226を含む。ディレクトリ220は、\companies\acmeディレクトリであって、ソースオブジェクトを含んでいる。ディレクトリ230は、ディレクトリ226の詳細な図である。これは、\companies\acmeディレクトリに含まれたファイルネームの、ファイルシステム識別子に対するマッピングを含む。ファイルシステム識別子は、ファイルシステムにおけるファイルを独特に識別する。例えば、ディレクトリ230は、ファイルネーム「report.doc」をファイルシステム識別子〈fsid1〉に対してマップし、そしてファイルネーム「ledger.xls」をファイルシステム識別子〈fsid2〉に対してマップする。次いで、ファイルシステム識別子テーブル240は、各ファイルシステム識別子をアクセス情報ブロックに対してマップする。各アクセス情報ブロックは、位置及び記憶媒体、即ちファイルを構成するデータを含む「セクター」のリストを含んでいる。例えば、ファイルシステム識別子テーブルは、ファイルシステム識別子〈fsid1〉からアクセス情報ブロック250へマップすると共に、ファイルシステム識別子〈fsid2〉からアクセス情報ブロック260へマップする。アクセス情報ブロック260は、ソースファイルを構成するセクターのリストを含む。アクセス情報ブロック260は、レファレンス263、レファレンス264及びレファレンス265を構成するための3つのレファレンスを含む。これらのレファレンスは、媒体270のセクター273、274及び275を各々参照する。

10

20

【0012】

オペレーティングシステムは、典型的に、ユーザがオブジェクトを移動したり又は再ネーム付けできるようにするコマンドを含んでいる。オブジェクト間のリンクをサポートするシステムにおいては、移動又は再ネーム付けコマンドは、移動又は再ネーム付けされるオブジェクトを参照するリンクのパスネームを更新するように拡張することができる。しかしながら、オペレーティングシステムは、オブジェクトを移動したり再ネーム付けしたりするためにユーザが使用するコピー及び削除コマンドも与える。ユーザは、オブジェクトを同じディレクトリにコピーしそしてコピー元オブジェクトを削除することによりオブジェクトを再ネーム付けすることができる。ユーザは、オブジェクトを異なるディレクトリにコピーしそしてコピー元オブジェクトを削除することによりオブジェクトを移動することができる。ユーザがコピー及び削除コマンドを用いてソースオブジェクトを移動又は再ネーム付けするときには、ソースオブジェクトに対するリンクを解明することができなくなる。

30

【0013】

図3A-3Cは、コピー及び削除コマンドを用いてソースオブジェクトを再ネーム付けするときに生じる問題を説明するブロック図である。図3Aにおいて、report.docオブジェクト301は、ledger.xlsオブジェクト303に対するリンク302を含んでいる。このリンクは、ledger.xlsオブジェクトを参照するために相対的なパスネームを使用する。この点においてリンクが解明された場合には、ledger.xlsオブジェクトに対して正しく解明することになる。図3Bにおいては、report.docオブジェクト、ledger.xlsオブジェクト及びリンクが不変である。しかしながら、ledger.xlsオブジェクトがgrowth.xlsオブジェクト304へコピーされている。この点において、リンクは、ledger.xlsオブジェクトを参照するパスネームをまだ含んでいるので、依然としてledger.xlsオブジェクトを解明する。図3Cにおいては、ledger.xlsオブジェクトが削除されている。リンクは、存在しないledger.xlsオブジェクトを依然として参照するので、リンクを解明することはできない。この点において、たとえgrowth.xlsオブジェクトが再ネーム付けされたledger.xlsオブジェクトであると意図されても、リンクの解明は失敗となる。ユーザがコピー及び削除コマンドを用いてソ

40

50

スオブジェクトを移動又は再ネーム付けするときには、ソースオブジェクトに対するいかなるリンクも解明できなくなる。

【0014】

ソースオブジェクトに対するリンクが不正確に解明される別の状況は、リンクを含むオブジェクトが異なるディレクトリに移動されたときに生じる。図4A-4Bは、コピー及び削除コマンドを用いてソースオブジェクトを移動するときに生じる問題を示すブロック図である。図4Aは、ソースledger.xlsオブジェクト403へのリンク402を含むreport.docオブジェクト401を示している。これらのreport.doc及びledger.xlsオブジェクトは、「\companies\acme」ディレクトリに含まれる。

10

【0015】

「\companies\ajax」ディレクトリは、異なるが同様のネームをもつledger.xlsオブジェクト404を含んでいる。report.docオブジェクトが「\companies\acme」ディレクトリにある間に、リンクは、ledger.xlsオブジェクト103に対して正しく解明する。図4Bは、「\companies\ajax」ディレクトリへ移動されたreport.docオブジェクトを示している。report.docオブジェクトが「\companies\ajax」ディレクトリにあるときは、リンクがledger.xlsオブジェクト404に対して不正確に解明する。リンクを含むオブジェクトが移動されて、そのリンクに記憶されたパスネームがオブジェクトを記述しないか又は間違ったオブジェクトを記述するときにも、同様の問題が生じる。

20

【0016】

リンクをサポートするあるコンピュータシステムにおいては、各リンクがパスネームではなくてオブジェクト識別子を含む。ロケータテーブルを用いてオブジェクト識別子がパスネームへとマップされる。ロケータテーブルにより加えられる間接化のレベルは、移動又は再ネーム付けされたソースオブジェクトに対してリンクを更新するプロセスを能率的なものにする。ソースオブジェクトに対していかに多くのリンクが存在しても、ロケータテーブルにおいてパスネームを一度変更するだけでそれらを全て更新することができる。図5はロケータテーブルの実施を示すブロック図である。ロケータテーブル501は、オブジェクト識別子511-514を含み、これらは各々ソースオブジェクトパスネーム521-524に対応する。ソースオブジェクト531-534の各々は独特のオブジェクト識別子541-544を含んでいる。ロケータテーブルは多くの異なるディレクトリにおいてオブジェクトに対するエントリーを含むので、絶対的なパスネームが使用される。ソースオブジェクトがリンクされるときには、そのオブジェクト識別子がリンクにコピーされる。オブジェクト識別子「1112」をもつオブジェクトは、オブジェクト識別子「1111」をもつオブジェクトに対してリンク551を含んでいる。オブジェクト識別子「1111」をもつオブジェクト531が移動又は再ネーム付けされた場合には、オブジェクト531の新たなネーム又は位置を正しく反映するようにパスネーム521を変更することによりリンクを保存することができる。

30

【0017】

ロケータテーブルの使用は効率を改善するが、リンクの維持に伴う新たな問題を生じる。媒体の欠陥又は意図しない削除の場合にオブジェクトが失われるのを防止するために、オリジナルオブジェクトが一次記憶装置（例えば、ハードディスク）から文書保存用記憶装置（例えば、フロッピーディスク）へしばしばコピーされる。このコピー動作をアーカイブと称し、そしてコピー動作により生じたオブジェクトをアーカイブされたオブジェクトと称する。アーカイブされたオリジナルオブジェクトが崩壊又は消去された場合には、それに対応するアーカイブされたオブジェクトを一次記憶装置へコピーして戻すことができる。このコピー動作をオブジェクトの回復と称し、そしてコピー動作により形成されたオブジェクトを回復されたオブジェクトと称する。回復されたオブジェクトは、通常、一次記憶装置の崩壊又は削除されたオブジェクトに取って代わる。しかしながら、ユーザは

40

50

オリジナルオブジェクトのネームを移動又は変更できるので、オブジェクトを回復すると、同じオブジェクトの2つのコピーが一次記憶装置上に生じることになる。同様に、ユーザはオブジェクトを異なるディレクトリに対して回復するように特定できるので、同じオブジェクトの2つのコピーが一次記憶装置上に生じることになる。両コピーは同じオブジェクト識別子を共用するが、ロケータテーブルにおいて個別のエントリーを有することもできる。同じオブジェクト識別子に対してロケータテーブルに2つのエントリーが存在し得るので、そのオブジェクト識別子からパスネームへのマッピングがあいまいなものとなる。その結果、複製されたオブジェクト識別子を含むリンクが、オリジナルオブジェクト又は回復されたオブジェクトのいずれかに対して解明される。オリジナルオブジェクト及び回復されたオブジェクトが互いに正確なコピーであるときには、このあいまいさが殆ど問題にならないが、いずれかのオブジェクトが変更されたときには、それらの共用オブジェクト識別子に対するリンクを解明するときに正しいオブジェクトを選択することが重要である。

10

【0018】

例えば、オブジェクト識別子「1111」を有するオブジェクト531が「\companies\acme」ディレクトリからアーカイブされそして「\companies\directory」へ回復される場合には、オブジェクト識別子「1111」及びパスネーム「\companies\ledger.xls」を含む新たなエントリー（図示せず）がロケータテーブルに形成される。

【0019】

20

532オブジェクトのリンクが解明されるときには、オブジェクト識別子「1111」をもつエントリーに対してテーブルをサーチするときに対応するロケータテーブルエントリーのどれに最初に遭遇するかに基づいて、オブジェクト識別子「1111」をもつオリジナルオブジェクトか又はオブジェクト識別子「1111」をもつ回復されたオブジェクトのいずれかに対して解明することができる。

【0020】

オブジェクトが厳密なコピーのままである場合は、リンクがどちらに対して解明するかは重要でない。しかしながら、オリジナルオブジェクトがより多くの情報を含むように編集される場合には、回復されたオブジェクトに対してリンクが解明するときに、オリジナルオブジェクトの編集により加えられる情報がクライアントオブジェクトに組み込まれない。

30

【0021】

ネットワークにより1つ以上の他のコンピュータシステムに接続されたコンピュータシステムの記憶装置に最初に記憶されたソースオブジェクトは、他の接続されたいずれかのコンピュータシステムの記憶装置へユーザが容易に移動することができる。上記の方法のいずれによってもソースオブジェクトが見つからない場合には、ソースオブジェクトをサーチするプログラムが、各接続されたコンピュータシステムへ要求を「放送」して、その記憶装置においてソースオブジェクトをサーチしそしてその結果を報告するのが一般的である。この「全処理サーチ(exhaustive search)」解決策は、ソースオブジェクトが識別可能であって且つ接続されたコンピュータシステムの記憶装置に記憶された場合にはこのソースオブジェクトを確かに有効に位置決めするが、この全処理サーチは、各記憶装置の全ての内容をサーチするのに各々の接続されたコンピュータシステムに対し著しい処理及び記憶検索リソースを必要とすると共に、要求を放送しそしてその結果を収集するために広範なネットワーク通信リソースを必要とするという点で非常に高価なものとなる。

40

【発明の開示】

【発明が解決しようとする課題】

【0022】

本発明が解決しようとする課題は、ソースオブジェクトと同じオブジェクト形成から派生するオブジェクトに対して解明することのできるソースオブジェクトへのリンクを確立するための方法及びシステムを提供することである。

50

【0023】

本発明の別の課題は、共通のオブジェクト形成から派生するオブジェクトを追跡するための方法及びシステムを提供することである。

【0024】

本発明の更に別の課題は、ソースオブジェクトと同じオブジェクト形成から派生するオブジェクトに対するリンクを解明するための方法及びシステムを提供することである。

【0025】

本発明の更に別の課題は、リンクのソースオブジェクトであるオブジェクトのためのアクセス可能な記憶ボリュームをサーチするための方法及びシステムを提供することである。

10

【課題を解決するための手段】

【0026】

これらの課題、及び本発明を以下に詳細に述べるにつれて明らかとなる他の目的、課題は、共通のオブジェクト形成から派生するオブジェクトを追跡しそしてそれに対するリンクを解明するための方法及びシステムによって達成される。好ましい実施形態では、システムがソースオブジェクトを形成する。次いで、システムはソースオブジェクトの形成を識別するために子孫識別子を発生する。次いで、システムはこの子孫識別子をソースオブジェクトに関連させる。好ましい実施形態では、システムは、ソースオブジェクトへのリンクが形成されるときだけ子孫識別子をソースオブジェクトに関連させる。後で、システムは、形成されたオブジェクトをコピーオブジェクトへとコピーする。ソースオブジェクトがコピーオブジェクトへコピーされると、システムは、ソースオブジェクトに関連した子孫識別子をコピーオブジェクトに関連させる。このように、コピーオブジェクトに関連した子孫識別子は、そのコピーオブジェクトがソースオブジェクトの形成から派生したことを指示する。又、システムは、好ましくは、ユーザが移動コマンドを特に使用せずにソースオブジェクトを有効に移動したときを検出し、そしてその有効に移動したオブジェクトに、オリジナルオブジェクトに関連した区別された子孫識別子に関連させることができる。

20

【0027】

システムは、ソースオブジェクトの子孫識別子を含むリンクをクライアントオブジェクトに記憶することによりクライアントオブジェクトをソースオブジェクトにリンクする。又、リンクは、ソースオブジェクトを、同じ子孫識別子を有する他のオブジェクトから区別するための情報も含んでいる。ソースオブジェクトに対するリンクを解明するときには、システムは、リンクに含まれた子孫識別子及び区別情報を選択する。次いで、システムは、その選択された子孫識別子及び区別情報をもつオブジェクトをサーチする。その選択された子孫識別子及び区別情報をもつオブジェクトが見つかったときには、システムは、その見つかったオブジェクトに対してリンクを解明する。その選択された子孫識別子及び区別情報をもつオブジェクトが見つからないときには、システムは、その選択された子孫識別子をもつオブジェクトを、その選択された区別情報にかわりなくサーチする。その選択された子孫識別子をもつオブジェクトが見つかったときには、リンクシステムは、その見つかったオブジェクトに対するリンクを解明する。

30

40

【0028】

リンクを解明するときには、システムは、好ましくは、最適な順序で一連のボリュームにおいてリンクのソースオブジェクトをサーチする。システムは、好ましくは、先ず、リンクに記憶されたパスネームをチェックし、次いで、暗示されるボリュームをサーチし、全てのローカルボリュームをサーチし、自動ボリュームリストにおけるボリュームをサーチし、手動ボリュームリストにおけるボリュームをサーチし、リモートボリュームリストのリストにより指示されるリモートボリュームリストにおけるボリュームをサーチし、そして全ての接続されたマシンへサーチ要求を放送する。又、好ましくは、システムは、パスネームにより指定されたディレクトリにおいてソースオブジェクトファイルネームをルックアップする段階をバイパスするように、リンクに使用されたオブジェクト識別子から

50

ファイルシステム識別子へ直接マップするオブジェクト識別子テーブルを具現化する。

【発明の効果】

【0029】

以上、本発明を好ましい実施形態を参照して説明したが、本発明の精神及び範囲から逸脱せずに、その形態及び細部に種々の変更又は修正がなされ得ることが当業者に理解されよう。例えば、ファシリティは、コンピュータシステムに記憶されないオブジェクト、例えば、物理的な文書又はアートワークの子孫を追跡することもできる。更に、異なる構成のオブジェクト識別子が子孫情報を伝えることもできる。

【発明を実施するための最良の形態】

【0030】

本発明の好ましい実施形態は、コンピュータシステムにおいてオブジェクトを追跡しそしてオブジェクトに対するリンクを解明するためのソフトウェアベースのオブジェクトリンク構成体（ファシリティ）である。このファシリティは、リンクのソースである各オブジェクトに、2つの要素、即ち子孫識別子及び区別識別子より成るオブジェクト識別子を指定する。オブジェクトは、2つの個別の方法で存在させることができ、即ちスクラッチから形成できるか又は既存のオブジェクトからコピーできるかのいずれかである。オブジェクトが形成されるときには、そのオブジェクトに、好ましい独特の子孫識別子及び区別識別子より成るオブジェクト識別子が指定される。オブジェクトがコピーされるときには、そのコピーオブジェクトに、そのコピー元（オリジナル）オブジェクトと同じ子孫識別子より成る好ましい独特のオブジェクト識別子が指定されると共に、その同じ子孫識別子をもつ他の各々のオブジェクト識別子の区別識別子とは異なる区別識別子が指定される。ファシリティは、ソースオブジェクトへのリンクを確立するときに、ソースオブジェクトのオブジェクト識別子をリンクに記憶する。リンクが解明されるときには、オリジナルのソースオブジェクトが依然としてアクセスできる場合には、ファシリティは、そのリンクに含まれた子孫識別子及び区別識別子をもつオブジェクトをサーチすることにより、オリジナルのソースオブジェクトを位置決めする。オリジナルのソースオブジェクトがアクセス不能であり、例えば、削除されている場合には、ファシリティは、リンクに含まれた子孫識別子をもつオブジェクトを、その選択された区別識別子に係わりなくサーチすることにより、オリジナルソースオブジェクトのコピーを見つけることができる。又、ファシリティは、好ましくは、ユーザがリンクのソースを第1ソースオブジェクトから第2ソースオブジェクトへ変更しようとする状況を検出し、そして全オブジェクト識別子を第1ソースオブジェクトから第2ソースオブジェクトへ自動的に転送する。これらの状況は、ユーザがコピー及び削除コマンドを用いてソースオブジェクトを有効に移動するとき、ユーザが典型的なアプリケーションプログラムを用いてソースオブジェクトの改定バージョンをセーブするとき、又はリンク及びリンクのソースの両方を含むオブジェクトのグループをユーザがコピーするときを含む。この後者の場合には、子孫識別子のみがコピーされ、そして新たな区別識別子が使用される。

【0031】

別の実施形態においては、子孫識別子及び区別識別子を含むのではなく、オブジェクト識別子の各々は、子孫識別子と、一連のコピー識別子とを含む。この実施形態では、ファシリティは、オブジェクトが形成されたときにオブジェクトに子孫識別子を指定し、次いで、オブジェクトがコピーされるたびにオブジェクトにコピー識別子を指定する。コピー識別子は、区別識別子として働き、同じ形成を分担するオブジェクト間を区別する。又、この実施形態では、オブジェクトの特定の継承に関する情報も追加される。これは、ファシリティが、オブジェクト識別子の分かっているオブジェクトの祖先と子孫の間を区別できるようにする。

【0032】

このように、オリジナルソースオブジェクトが使用できないリンクを解明するときには、ファシリティは、オリジナルソースオブジェクトの子孫、特に、近い子孫であるオブジェクトへの解明に優先順位を与えることができる。

10

20

30

40

50

【0033】

ファシリティがリンクを効率的に解明できるのが望ましい。このため、ファシリティは、ボリューム（例えば、ディスクドライブ）に記憶されたオブジェクトに対する全てのオブジェクト識別子を、その各々を対応するソースオブジェクトの位置にマップするオブジェクト識別子テーブルに記憶するのが好ましい。解明中に、ファシリティは、ソースオブジェクトが記憶されたファイルシステムをサーチすることなく、オブジェクト識別子テーブルを用いてソースオブジェクトを直接参照する。又、ファシリティは、リンクの解明に使用するためにソースオブジェクトをおそらく含むリモートボリュームのリストを維持するためにリモートボリュームへの接続も監視する。更に、ファシリティは、ソースオブジェクトが更新されるときにソースオブジェクトの識別子が関連性を断つのを防止する。これは、コピーされそして削除されるオブジェクトの「短命」リストを維持しそしてその削除されたオリジナルオブジェクトの識別子をコピーに指定することにより行われる。更に、ファシリティは、最適なサーチ戦略を用いて、クライアントオブジェクトが記憶されたボリューム以外のボリュームに記憶されたソースオブジェクトを位置決めする。この戦略によれば、ファシリティは、先ず、クライアントオブジェクトの位置に対してソースオブジェクトの位置を記述するリンクに記憶された相対的なパスにより指定されたディレクトリをサーチする。次いで、ファシリティは、最後にリンクが解明されたときにソースオブジェクトを含むボリュームをサーチする。

10

【0034】

図6は、ファシリティがその中で好ましく動作するところの汎用コンピュータシステムの高レベルブロック図である。コンピュータシステム601は、中央処理ユニット（CPU）602と、コンピュータメモリ（メモリ）603と、入力／出力装置604とを備えている。入力／出力装置の中には、ハードディスクドライブのような記憶装置605と、ネットワーク接続部606がある。オブジェクトは、メモリ又は記憶装置のいずれに存在することもできる。更に、オブジェクトは、コンピュータシステム間でネットワーク接続部を経てコピー、移動又はアクセスすることができる。

20

【0035】

オブジェクトがスクラッチから形成されるときには、ファシリティは、それに新たな子孫識別子及び新たな区別識別子を指定する。子孫識別子は、どのオブジェクトが同じ形成から派生するか、即ちどのオブジェクトがその形成されたオブジェクトのコピーでありそしてこれらコピーされたオブジェクトのコピーであるか、等々を反映するように意図される。好ましい実施形態では、子孫識別子は、16バイト数字である。新たな子孫識別子は、日付、時刻及びコンピュータシステムの識別情報をエンコードすることにより発生するのが好ましい。これは、使用する時間の表示が十分に正確である限り、通常は、厳密に同じ日付及び時刻に同じコンピュータシステムにおいて2つのオブジェクトを形成することが不可能であるから、子孫識別子を独特のものにする有効な方法である。コピー識別子も同様に発生されるのが好ましい。区別識別子は、同じ子孫識別子を有するオブジェクト、即ち同じ形成から派生するオブジェクトを区別するよう意図される。

30

【0036】

好ましい実施形態では、区別識別子は、4バイト数字である。区別識別子として更に大きい数字を使用してもよいが、4バイト数字は、オブジェクトのコピーを区別するための区別識別子の容量と、区別識別子を記憶するための記憶リソースの消費量との間の有効な妥協であることが分かっている。オブジェクトについてなされるコピーの回数は、通常は、比較的僅かであるから、区別識別子が同じオブジェクトのコピーを区別するために非常に大きな容量をもつ必要はない。ランダム数字を選択することにより新たな区別識別子が発生されるのが好ましい。或いは又、動作中のコンピュータシステムを用いて新たな区別識別子が発生されてもよい。以下で詳細に述べるように、付加的な記憶量を各々消費する使用されるオブジェクト識別子の個数を制限するために、最初にオブジェクトへのリンクが形成されるときにオブジェクト識別子がオブジェクトに対して形成されそしてオブジェクトに関連されるのが好ましい。

40

50

【0037】

オブジェクトがコピーされるときには、ファシリティは、オリジナルオブジェクトの子孫識別子及び新たな区別識別子をコピーオブジェクトに指定する。オリジナルオブジェクト及びコピーオブジェクトは、共通のオブジェクト形成から派生すると言える。子孫識別子は、オブジェクトに関連していることを示し、そして区別識別子は、オブジェクトを区別するために使用される。オブジェクトがアーカイブされそして回復されるときには、それがコピーされたと考えられ、ファシリティは、それに新たな区別識別子を指定する。オブジェクトが移動又は再ネーム付けされるときは、その子孫識別子及びその区別識別子の両方が同じままである。

【0038】

ファシリティがオブジェクトに対してオブジェクト識別子（子孫識別子及び区別識別子を含む）を発生するときには、そのオブジェクト識別子にオブジェクトに関連させ、（a）ユーザがオブジェクトへのリンクを確立すると決定したときにファシリティがオブジェクト識別子を含むリンクを確立できるようにし、そして（b）ファシリティは、リンクを解明するときに、あるオブジェクト識別子を有するオブジェクトをサーチすることができる。好ましい実施形態においては、ファシリティがオブジェクト識別子をオブジェクトに関連させるときに、オブジェクト識別子をオブジェクト内に記憶する。このようにして、オブジェクトは、それ自身のオブジェクト識別子を知る。ファシリティは、オブジェクト識別子に関連オブジェクトのパスネームにマップさせるオブジェクト識別子テーブルを確立するのが好ましい。オブジェクトに対してオブジェクト識別子が発生されるときには、ファシリティは、オブジェクトのパスネームに対するそのオブジェクト識別子のマップを含むようにオブジェクト識別子テーブルを更新する。ユーザがオブジェクトを移動するか又は再ネーム付けする場合に、ファシリティは、オブジェクト識別子テーブルに記憶されたパスネームを更新する。テーブルにオブジェクト識別子を記憶することにより、ファシリティはオブジェクトを迅速にサーチすることができる。

【0039】

図7は、オブジェクト識別子がオブジェクトパスネームに対してマップされたオブジェクト識別子テーブルの具現化を示す概略図である。この図は、report.docクライアントオブジェクト700を示し、これは、ledger.xlsソースオブジェクトへのリンク701を含んでいる。このリンクはソースオブジェクトを独特に識別するオブジェクト識別子702（〈objid1〉）を含んでいる。更に、この図は、オブジェクト識別子をファイルパスネームへとマップするオブジェクト識別子テーブル710を示している。ファシリティは、リンク701を解明しようとするときに、オブジェクト識別子テーブルを読み、オブジェクト識別子〈objid1〉711がパス「\companies\acme\ledger.xls」712へマップすることを決定する。更に、この図は、ルートディレクトリ720及び他のディレクトリ721-726より成るファイルシステムハイアラキも示している。ソースオブジェクトのパス712は、「\companies\acme」ディレクトリ726を特定し、これは、ディレクトリ730として詳細に示されている。ディレクトリ730は、ファイルネームからファイルシステム識別子へのマップを含み、これらファイルシステム識別子は、次いで、ファイルシステム識別子テーブル740により、オブジェクトがボリュームの媒体770から読み出せるようにするアクセス情報を含むアクセス情報ブロックへとマップされる。例えば、「report.doc」ファイルネーム731はファイルシステム識別子733（〈fsid1〉）へマップされ、そしてソースオブジェクトに対応する「ledger.xls」ファイルネームは、ファイルシステム識別子734（〈fsid2〉）へマップされる。ファイルシステム識別子テーブル740はファイルシステム識別子からアクセス情報ブロックへマップする。例えば、ファイルシステム識別子741（〈fsid1〉）はアクセス情報ブロック750へマップされ、これは、対応するオブジェクトが媒体770のセクター771及び772を占有することを示す。同様に、ファイルシステム識別子742（〈fsid2〉）はアクセス情報ブロック760へマップされ、これは、ソースオブジェクト

10

20

30

40

50

が媒体 770 のセクター 773、774 及び 775 を占有することを示す。次いで、ソースオブジェクトを検索し、そしてセクター 773、774 及び 775 を検索することによりリンクを完全に解明することができる。

【0040】

好ましい別の実施形態において、ファイルシステム識別子テーブルがファイルシステムによって露出される場合には、オブジェクト識別子をパスネームではなくてファイルシステム識別子へ直接マップするようなオブジェクト識別子テーブルをファシリティが維持して使用し、これにより、パスネームにより指定されたディレクトリにおけるファイルネームをルックアップする段階をバイパスするのが効率的である。図 8 は、オブジェクト識別子がファイルシステム識別子へ直接マップされたオブジェクト識別子テーブル 880 の具

10

【0041】

オブジェクト識別子 802 (〈objid1〉) を含むリンク 801 を解明するために、ファシリティは、オブジェクト識別子テーブル 880 において「〈objid1〉」をルックアップする。このオブジェクト識別子は、ファイルシステム識別子「〈fsid2〉」882 へマップされ、従って、ファシリティは、ファイルシステム識別子テーブル 840 においてファイルシステム識別子「〈fsid2〉」をルックアップする。ファイルシステム識別子「〈fsid2〉」はアクセス情報ブロック 860 へマップされ、これは、ソースオブジェクトが媒体 870 のセクター 873、874 及び 875 を占有することを示す。次いで、ソースオブジェクトを検索し、そしてセクター 873、874 及び 87

20

【0042】

ハイブリッドファイルシステムにおいては、各ソースオブジェクトに対するファイルシステム識別子をオブジェクト識別子テーブルに記憶するために使用できるが、ソースオブジェクトに対応するセクターを検索するための機構であって、そのファイルシステム識別子を用いて検索されるべきソースオブジェクトを識別するような機構は存在しない。むしろ、ソースオブジェクトに対応するセクターを検索する唯一の機構は、検索されるべきオブジェクトを識別するために絶対パスネームを使用するようなオブジェクト検索のためのファイルシステムサービスである。このようなハイブリッドファイルシステムにおいて、ファシリティは、好ましくは、ファイルシステム識別子を用いて、検索されるべきソース

30

【0043】

オブジェクトは、ボリュームに記憶されてもよい。ボリュームとは、特定の記憶装置であるか又は特定の記憶装置の区画である。各ボリュームは、それ自身のルートディレクトリ及びファイルシステムハイアラキを有している。コンピュータシステムは、それ自身の多数のボリュームを有し、そして更に、ネットワークコネクタを経て他のコンピュータシステムのボリュームへアクセスすることができる。好ましい実施形態では、ファシリティは各ボリュームに含まれたオブジェクトに対し個別のオブジェクト識別子テーブルを確立する。このテーブルは、それがオブジェクト識別子を含んでいるところのボリュームに記憶されるのが好ましい。これは、参照されたオブジェクトがアクセスできる場合及びその場合にのみ、オブジェクト識別子テーブルをアクセスできるよう確保する。一方、オブジェクト識別子テーブルを、それがオブジェクト識別子を含むところのボリュームに記憶できないこともある。ファシリティは、例えば、ボリュームが読み取り専用であるか又は

40

50

それがいっぱいであるために、ボリュームに書き込みできないことがある。このようなボリュームに対するオブジェクト識別子テーブルは、別のボリュームに記憶することができる。

【0044】

別の好ましい実施形態においては、ファシリティは、関連するオブジェクトが最初にリンクされるときにオブジェクト識別子テーブルにオブジェクト識別子のみを追加する。又、ファシリティは、オブジェクトが最初にリンクされるまで、オブジェクト識別子とオブジェクトとの関連を抑制することもできる。これは、オブジェクト識別子が決して使用されないとき、即ちオブジェクトが決してリンクされないときに、オブジェクト識別子を発生して記憶するオーバーヘッドを排除するという利点を有する。しかしながら、これは、ある関連オブジェクトへのリンクを確立して解明するファシリティの能力を排除するという欠点も有する。

10

【0045】

例えば、オブジェクトが最初にコピーされ、リンクされそして削除される場合には、コピーオブジェクトに対してリンクを解明することができない。

【0046】

ファシリティがソースオブジェクトへのリンクを確立するたびに、ファシリティは、ソースオブジェクトのオブジェクト識別子をリンクへコピーする。好ましい実施形態においては、ファシリティは、ソースオブジェクトのパスネーム又は少なくともそのボリュームネームをリンクに記憶する。ソースオブジェクトが移動、コピー又は再ネーム付けされるときに、たとえこのパスネーム又はボリュームネームが更新されなくても、オブジェクトをサーチすべきスタート点を確立することができる。

20

【0047】

図9は、ソースオブジェクトを位置決めするための第1のMatchObjectサブルーチンを示すフローチャートである。このサブルーチンは、先ず、全オブジェクト識別子が一致するオブジェクトをサーチし、次いで、必要に応じて子孫識別子が一致するオブジェクトをサーチする。サブルーチンは、オブジェクト識別子をパラメータとして受け取る。ファシリティは、サブルーチンを呼び出すときに、それが解明するリンクに記憶されたオブジェクト識別子を通す。サブルーチンは、一致するオブジェクト識別子をもつオブジェクトに対し全てのオブジェクトをサーチする。このサーチは、オリジナルサーチオブジェクトが依然としてアクセスできる場合にこのオブジェクトのみを見つける。ステップ902において、一致するオブジェクト識別子をもつオブジェクトが見つかった場合に、サブルーチンは、ステップ903に続き、さもなくば、ステップ904へ続く。

30

【0048】

ステップ903において、サブルーチンは、一致するオブジェクト識別子をもつオブジェクトへのリンクを解明する。次いで、サブルーチンは、復帰となる。

【0049】

ステップ904において、サブルーチンは、一致する子孫識別子をもつオブジェクトに対し全てのオブジェクトをサーチする。このサーチは、オリジナルソースオブジェクトと同じ形成から派生するオブジェクトを見つける。ステップ905において、一致する子孫識別子をもつ2つ以上のオブジェクトが見つかった場合には、サブルーチンは、ステップ906へ続き、さもなくば、サブルーチンはステップ909へ続く。

40

【0050】

ステップ906ないし907において、サブルーチンは、リンクを解明すべき関連オブジェクトをユーザが選択できるようにする。ステップ906において、サブルーチンは、一致する子孫識別子をもつオブジェクトのリストを表示する。

【0051】

ステップ907において、サブルーチンは、ユーザがリストからオブジェクトを選択できるようにする。ステップ908において、サブルーチンは、ユーザにより選択されたオブジェクトへのリンクを解明する。ステップ908及び910では、ファシリティが、同

50

じ子孫識別子及び異なる区別識別子を有するオブジェクトへのリンクを解明するときに、ファシリティは、リンクに記憶された区別識別子を解明されるオブジェクトの区別識別子へと変更するのが好ましい。これは、次に必要になったときにこのオブジェクトへのリンクをファシリティが更に容易に解明できるようにする。次いで、サブルーチンは復帰する。

【0052】

ステップ909において、一致する子孫識別子を有するオブジェクトが見つかった場合に、サブルーチンはステップ910へ続き、さもなければ、サブルーチンはステップ911に続く。ステップ910では、サブルーチンは、一致する子孫識別子をもつオブジェクトへのリンクを解明する。次いで、サブルーチンは復帰する。ステップ911では、オブジェクトへのリンクの解明に失敗し、復帰となる。ファシリティは、特定のオブジェクトへのリンクを解明すると、そのオブジェクトをリンクのソースオブジェクトとして指定すると言える。

10

【0053】

別の実施形態においては、ファシリティは、MatchObjectサブルーチンの変更バージョンを呼び出す。図10は、ソースオブジェクトを位置決めするための第2のMatchObjectサブルーチンのフローチャートである。

【0054】

この変更サブルーチンは、最初に、一致する子孫識別子を有する全てのオブジェクトをサーチし、次いで、これらオブジェクトの中で、一致する区別識別子を有するオブジェクトをサーチする。又、この変更サブルーチンは、オブジェクト識別子をパラメータとして受け取る。

20

【0055】

ステップ1001において、変更サブルーチンは、一致する子孫識別子をもつオブジェクトに対し全てのオブジェクトをサーチする。ステップ1002において、変更サブルーチンは、一致する区別識別子をもつオブジェクトに対し、ステップ1001で見つかった全てのオブジェクトをサーチする。ステップ1003において、一致する区別識別子をもつオブジェクトが見つかった場合には、変更サブルーチンは、ステップ1004へ続き、さもなければ、ステップ1005へ続く。ステップ1004において、変更サブルーチンは、一致する区別識別子をもつオブジェクトへのリンクを解明する。このオブジェクトは、オリジナルソースオブジェクトである。次いで、変更サブルーチンは、復帰となる。

30

【0056】

ステップ1005において、一致する子孫識別子をもつ2つ以上のオブジェクトが見つかった場合には、変更サブルーチンはステップ1006へ続き、さもなければ、ステップ1009へ続く。ステップ1006ないし1007において、変更サブルーチンは、リンクを解明すべきところの一致する子孫識別子をもつオブジェクトをユーザが選択できるようにする。一致する子孫識別子をもつオブジェクトは、オリジナルソースオブジェクトと同じ形成から派生するオブジェクトである。ステップ1006において、変更サブルーチンは、一致する子孫識別子をもつオブジェクトのリストを表示する。ステップ1007において、変更サブルーチンは、ユーザがリストからオブジェクトを選択できるようにする。ステップ1008では、変更サブルーチンは、ユーザにより選択されたオブジェクトへのリンクを解明する。ステップ1008及び1010では、同じ子孫識別子及び異なる区別識別子を有するオブジェクトへのリンクをファシリティが解明するときに、ファシリティは、リンクに記憶された区別識別子を解明されるオブジェクトの区別識別子へと変更するのが好ましい。これは、次に必要になったときにこのオブジェクトへのリンクをファシリティが更に容易に解明できるようにする。次いで、変更サブルーチンは復帰する。

40

【0057】

ステップ1009において、一致する子孫識別子を有するオブジェクトが見つかった場合に、変更サブルーチンはステップ1010へ続き、さもなければ、変更サブルーチンはステップ1011に続く。ステップ1010では、変更サブルーチンは、一致する子孫識別

50

子をもつオブジェクトへのリンクを解明する。これはオリジナルソースオブジェクトに関連したオブジェクトである。次いで、変更サブルーチンは復帰となる。

【0058】

ステップ1011では、変更サブルーチンは、オブジェクトへのリンクの解明に失敗する。次いで、変更サブルーチンは復帰となる。

【0059】

いずれかのMatchObjectサブルーチンを使用することにより、ファシリティは、コピー及び削除コマンドを用いてソースオブジェクトを移動又は再ネーム付けするときに生じる問題を克服することができる。図11Aないし11Cは、コピー及び削除コマンドを用いてソースオブジェクトを移動又は再ネーム付けするときに生じる問題をファシリティがいかに克服するかを示すブロック図である。この例及びこれ以降の例の説明上、子孫識別子及び区別識別子の両方は4つの10進数に短縮する。図11Aにおいて、report.docオブジェクト1101は、ソースオブジェクトledger.xls1103へのリンク1102を含んでいる。このリンクは、ledger.xlsオブジェクトの子孫識別子「7717」及びledger.xlsオブジェクトの区別識別子「2155」を含んでいる。このときリンクが解明された場合は、ファシリティは、先ず、同じ子孫識別子及び区別識別子を有するオブジェクトをリンクとしてサーチする。第1のサーチは、ledger.xlsオブジェクトを見つけ、そしてファシリティは、ledger.xlsオブジェクトに対して解明する。図11Bは、report.doc及びledger.xlsオブジェクトと、不変のリンクとを示している。ledger.xlsオブジェクトのコピー1104が「growth.xls」というネームで形成される。ledger.xlsオブジェクトのコピーとして、growth.xlsオブジェクトは、同じ子孫識別子「7177」を保持し、そして新たな区別識別子「3216」を受け取る。

【0060】

この点において、リンクは、依然としてledger.xlsオブジェクトに対して解明する。というのは、同じ子孫識別子及び同じ区別識別子をリンクとして有するからである。図11Cでは、ledger.xlsオブジェクトが削除されている。この点においてリンクが解明された場合には、ファシリティは、同じ子孫識別子及び区別識別子をリンクとして有するオブジェクトを最初にサーチする。このようなオブジェクトはもはや存在しないので、第1のサーチは失敗となる。次いで、ファシリティは、同じ子孫識別子をリンクとして有するオブジェクトをサーチする。この第2のサーチは、リンクが解明される場所のgrowth.xlsオブジェクトを見つける。ファシリティは、この状態においてgrowth.xlsオブジェクトに対して解明できるので、コピー及び削除の問題を受けない。

【0061】

好ましい実施形態では、ファシリティは、コピー及び削除問題を更に効率的に解消するために移動確認ログを具現化する。ユーザがオブジェクトのコピーを行うときには、ファシリティは、短時間コピーログとも称する移動確認ログにコピーの指示を記憶する。ログに記憶されるコピーの指示は、オリジナルオブジェクト及びコピーオブジェクトの共通の子孫識別子と、形成されたオブジェクト及びコピーオブジェクトの区別識別子とを含むのが好ましい。ログは、少数のエントリーのみのスペースを有する円形リストである。それ故、いかなる所与の時間においても、ログは、ユーザにより行われた最後の若干のコピーのみの指示を保持する。ユーザがオブジェクトを削除するときには、ファシリティは、削除されたオブジェクトがコピーされたという指示に対してログをサーチする。ログがこのような指示を含む場合に、ファシリティは、コピーオブジェクトの区別識別子を削除されたオブジェクトの区別識別子へと変更し、削除されたオブジェクトに対して確立された全てのリンクがコピーオブジェクトに対して解明するようにさせる。この実施形態では、ファシリティは、多くの場合に、ファシリティの処理の集中するラインサーチ方法を用いることなくコピー及び削除問題を解消することができる。しかしながら、オリジナルオブジ

ェクトのコピー動作と削除との間のコピーオブジェクトに対して確立されるリンクは、無効となる。

【0062】

又、ファシリティは、クライアントオブジェクトを異なるディレクトリへ移動する問題も克服する。図12A-12Bは、クライアントオブジェクトを異なるディレクトリに移動する問題をファシリティがいかに克服するかを示すブロック図である。図12Aは、report.docオブジェクト1201を示し、これは、ソースledger.xlsオブジェクト1203へのリンク1202を含む。このリンクは、ledger.xlsオブジェクトの子孫識別子「7177」及びその区別識別子「2155」を含んでいる。report.doc及びledger.xlsオブジェクトは、「\companies 10
s\acme」ディレクトリに含まれる。「\companies\ajax」ディレクトリは、異なるが同様のネームをもつledger.xlsオブジェクト1204を含んでいる。「\companies\ajax」ディレクトリにおけるledger.xlsオブジェクトは、独立して形成されたものであるから、「\companies\acme」ディレクトリのledger.xlsオブジェクトとは異なる子孫識別子及び異なる区別識別子を有する。report.docオブジェクトが「\companies\acme」ディレクトリにある間にリンクが解明される場合に、ファシリティは、子孫識別子及び区別識別子がリンクにおけるものと一致するようなオブジェクトに対してボリューム内の全てのオブジェクトをサーチする。このサーチは、「\companies\acme」ディレクトリにおいてledger.xlsオブジェクトを見つける。図12B 20
は、「\companies\ajax」ディレクトリへのreport.docオブジェクトの移動を示している。report.docオブジェクトが「\companies\ajax」ディレクトリにある間にリンクが解明される場合には、ファシリティは、リンクと同じ子孫識別子及び区別識別子を有するオブジェクトに対しボリューム内の全てのオブジェクトをサーチする。「\companies\acme」ディレクトリが「\companies\ajax」ディレクトリと同じボリューム内にある限り、ファシリティは、ledger.xlsオブジェクト及び「\companies\acme」ディレクトリへのリンクを見つけて解明する。ファシリティは、リンクを含むオブジェクトが移動された後もリンクを正しく解明できるので、ファシリティは、クライアントオブジェクトを異なるディレクトリに移動する問題を受けない。 30

【0063】

更に、ファシリティは、オブジェクトの新たな改定をセーブするために多くのアプリケーションプログラムによって使用されるプロセスにより生じるリンク追跡の問題も克服する。オブジェクトの以前の改定をオブジェクトの現在の改定と単に置き換えるのではなく、これらのアプリケーションは、オブジェクトの現在の改定を一時的なオブジェクトに記憶し、次いで、以前の改定を現在オブジェクトネームからバックアップオブジェクトネームへ再ネーム付けし、そして一時的なオブジェクトを一時的なオブジェクトネームから現在オブジェクトネームへ再ネーム付けする。図13は、オブジェクトの新たな改定をセーブするために多くのアプリケーションプログラムが使用するプロセスを示すオブジェクト状態タイミング図である。このオブジェクト状態タイミング図は、アプリケーションプロ 40
グラムによって形成される原簿オブジェクトの次々の改定に各々対応する3つのオブジェクト1301-1303の状態を示している。これらの状態は、次々に遅い時間t1-t6を含む時間軸1304に対して示されている。時間t1に、Revision1及びRevision2が存在する。Revision2オブジェクトは、ネーム「ledger.xls」、オブジェクト識別子「0002:0001」を有し、ユーザは、これを原簿オブジェクトの「現在」改定即ち最後にセーブされた改定であると考え。Revision1オブジェクトは、ネーム「ledger.bak」、オブジェクト識別子「0001:0001」を有し、ユーザは、これを原簿オブジェクトの「バックアップ」改定即ち最後から2番目にセーブされた改定であると考え。時間t1に、ユーザは、原簿オブジェクトの現在改定、Revision2に対するリンクを形成する（図示せず）。このリ 50

リンクは、Revision 2 に対するオブジェクト識別子「0002:0001」を含む。時間 t2 に、アプリケーションプログラムは、原簿オブジェクトの最新の改定である Revision 3 をセーブする処理を開始する。アプリケーションは先ず Revision 3 オブジェクトを形成する。Revision 3 オブジェクトは、最初に、一時的なファイルネーム「ledger.tmp」及びオブジェクト識別子「0003:0001」を有する。時間 t3 において、アプリケーションは、Revision 1 オブジェクトを削除する。時間 t4 において、アプリケーションは、Revision 2 オブジェクト「ledger.bak」と再ネーム付けする。Revision 2 オブジェクトのオブジェクト識別子は変化しない。時間 t5 において、アプリケーションは、Revision 3 オブジェクト「ledger.xls」と再ネーム付けする。Revision 3 オブジェクトのオブジェクト識別子は変化しない。この点において、Revision 3 オブジェクトは原簿オブジェクトの現在改定であり、そして Revision 2 オブジェクトは原簿オブジェクトのバックアップ改定である。殆どの場合に、ユーザは、ソースオブジェクトへのリンクがソースオブジェクトの現在改定に対して解明されることを期待すると仮定する。しかしながら、時間 t1 で形成されたリンクが時間 t6 で解明される場合には、リンクは、Revision 2 (バックアップ) オブジェクトに対して解明される。というのは、異なるオブジェクト識別子を有する Revision 3 (現在) オブジェクトではなくて、リンクに記憶されたオブジェクト識別子を有するからである。

【0064】

図14は、オブジェクトの新たな改定をセーブするために多くのアプリケーションプログラムが使用するプロセスをいかにファシリティが好ましく取り扱うかを示したオブジェクト状態タイミング図である。時間 t4 において、Revision 2 オブジェクトが再ネーム元のネーム「ledger.xls」から再ネーム先のネーム「ledger.bak」へ再ネーム付けされるときに、再ネーム元のネームは「消失する」と言える。即ち、「消失される」ネームを有するオブジェクトは停止される。又、ネームを有するオブジェクトが削除されるときにもネームが消失することがある。ファシリティは、消失されるネーム、この場合は「ledger.xls」を最新のネーム消失ログへ追加するために Process Disappearance サブルーチンを呼び出すのが好ましい。

【0065】

図15は、Process Disappearance サブルーチンのフローチャートである。このサブルーチンは、オブジェクトネームが消失するたびに呼び出される。ステップ1501において、ファシリティは、消失するネームと、消失するネームを有するオブジェクトのオブジェクト識別子と、消失したネームが再び現れるときにオブジェクトに転送されるオブジェクトに対応する他の情報とを含むネーム消失ログへエントリーを付加する。時間 t5 において、Revision 3 オブジェクトが「ledger.xls」と再ネーム付けされるときに、このネームは「出現する」と言える。ネームを有するオブジェクトが形成されるとき又は既存のオブジェクトがネームを有するように再ネーム付けされるときに、ネームが出現する。ネームが出現すると、ファシリティは、ネームが「再出現」するかどうか、即ち最近消失した後に現れるかどうかを決定する。出現するネームが再出現するかどうかを決定するために、ファシリティは、Process Appearance サブルーチンを呼び出して、出現するネームに対し最新ネーム消失ログをチェックする。最新ネーム消失ログが出現するネームを含む場合には、ファシリティは、ネームが再出現すると結論する。ファシリティは、ネームが再出現すると結論する場合に、再出現するネームを有するオブジェクトを消失したネームを有するオブジェクトとユーザが置き換えようとするという仮定を使用し、そして消失したネームを有するオブジェクトへのリンクが再出現するネームを有するオブジェクトに対して解明されるようにする。このため、再出現するネームを有するオブジェクトがまだオブジェクト識別子を有していない場合に、ファシリティは、消失したネームを有するオブジェクトのオブジェクト識別子を再出現するネームを有するオブジェクトへ「転送」又は「トンネル通過」させ、再出現するネームを有するオブジェクトのオブジェクト識別子を消失したネームを有するオブジェク

10

20

30

40

50

トのオブジェクト識別子に変更するのが好ましい。(消失したネームを有するオブジェクトのオブジェクト識別子は、この目的で最新ネーム消失ログに記憶されるのが好ましい。)

【0066】

図16は、Process Appearanceサブルーチンのフローチャートである。このProcess Appearanceサブルーチンは、オブジェクトネームが現れるたびに呼び出される。ファシリティは、先ず、出現するネームがネーム消失ログにあるかどうか決定する。ステップ1601において、出現するネームがネーム消失ログにある場合には、出現するネームは再出現し、ファシリティは、ステップ1602へ続き、さも
10
なくば、出現するネームは再出現せず、ファシリティは復帰となる。ステップ1602において、ネームが再出現するオブジェクトがオブジェクト識別子を既に有している場合には、ファシリティは復帰となり、さもなくば、ファシリティはステップ1603へ続く。ステップ1602は、オブジェクトのオリジナルオブジェクト識別子と別のものの置き換えを防止し、従って、オリジナルオブジェクト識別子へのリンクが遮断されることはない。ステップ1603では、ファシリティは、再出現するネームが消失したところのオブ
20
ジェクトのオブジェクト識別子を、ネームが再出現するところのオブジェクトヘトンネル通過させる。又、ステップ1603は、再出現するネームが消失したところのオブジェクトの他の特性、例えば、特徴、属性及び長いオブジェクトネームを、ネームが再出現するところのオブジェクトヘコピーすることを含むのが好ましい。この例では、Revision 3オブジェクトのオブジェクト識別子が時間t5において「0002:0001」へ
20
変更される。消失したネームを有するオブジェクトが依然として存在する(即ち、再ネーム付けされ、削除されていない)場合には、ファシリティは、オブジェクトに新たなオブジェクト識別子を指定するのが好ましい。新たなオブジェクト識別子は、同じ子孫識別子及び異なる区別識別子を有するのが好ましい。この例では、Revision 2オブジェクトのオブジェクト識別子が時間t5において「0002:0002」へ変更される。t1に形成されたリンクはオブジェクト識別子「0002:0001」を含むので、t5においてリンクはRevision 3オブジェクトに対して解明される。Revision 3オブジェクトが削除されるか又はアクセス不能なボリュームへ移動される場合には、リンクは、同じ子孫識別子及び異なる区別識別子を有するRevision 2オブジェクトに対して解明される。又、ファシリティは、消失したネームを有するオブジェクトに関連した他の特性、例えば、特徴、属性及び長いオブジェクトネームを、再出現するネームを有
30
するオブジェクトヘコピーするのが好ましい。

【0067】

ネームの出現は、ファイルシステムにおいてしばしば生じ、ネーム消失ログに記憶された消失したネームのリストをファシリティがしばしばサーチすることを必要とする。このサーチが行われるたびに消費される処理及びメモリアクセスリソースを最小にするために、ファシリティは、回転する消失ネームハッシュリストを用いて、最新ネーム消失ログが出現するネームを含むかどうかの決定を促進することが好ましい。回転する消失ネームハッシュリストは、少数のエントリーを含み、その各々は、最近消失したネームに対応し、そしてハッシュアルゴリズムにより例えば16ビットの小さな値に減少されたネームで構成される。消失するネームが最新ネーム消失ログに追加されるたびに、最も古いエントリーの内容が、その消失したネームに対し、減少された即ち「ハッシュ」された値と置き換えられる。ファシリティは、出現するネームが再出現するかどうか判断するよう試みるときに、最初に、出現するネームをハッシュし、そしてそれにより得られるハッシュ値に対し回転する消失ネームハッシュリストをチェックする。ハッシュ値に対し回転する消失ネームハッシュリストをチェックすることは、小さなハッシュ値を少数の他の小さなハッシュ値と比較することを含むので、チェックは非常に迅速に行われる。回転する消失ネームハッシュリストがハッシュ値を含まない場合には、ファシリティは、出現するネームが再出現しないと結論し、さもなくば、ファシリティは、出現するネームに対し最新ネーム消失リストをチェックするように進む。

10

20

30

40

50

【0068】

更に、ファシリティは、ソースオブジェクトをアーカイブ及び回復することに伴う問題も解消する。アーカイブされたオブジェクトが回復されるときには、ファシリティは、その子孫識別子を変更せず、新たな区別識別子を指定する。その結果、回復は、既存のオブジェクトと同一のオブジェクト識別子をもつ別のオブジェクトを導入することはない。リンクがオリジナルソースオブジェクトに対して形成されそしてオリジナルソースオブジェクトがその後にアーカイブされて回復される場合には、回復されたソースオブジェクトは、オリジナルソースオブジェクトと同じ子孫識別子、及び異なる区別識別子を有する。次いで、リンクが解明される場合には、ファシリティは、先ず、一致する子孫識別子及び区別識別子をもつオブジェクトを探す。オリジナルソースオブジェクトが依然として存在する場合には、このサーチによりそれが見つかる。オリジナルソースオブジェクトがもはや存在しない場合には、ファシリティは、同じ子孫識別子を有するオブジェクトをサーチする。この第2のサーチは、回復されたソースオブジェクトを見つけ、ファシリティはこれに対して解明する。アーカイブ及び回復は、同じオブジェクト識別子を有する2つの異なるオブジェクトを生じることではないので、ファシリティは、ソースオブジェクトのアーカイブ及び回復に伴う問題を受けない。

10

【0069】

更に、ファシリティは、ファイルのグループをコピーするときに遭遇するリンクの解明の問題を克服する。多くのファイルシステムは、オブジェクトのグループ、例えば、特定のディレクトリにおける全てのオブジェクト、又はファイルシステムハイアラーキのサブツリーにおける全てのオブジェクトを同時にコピーするコマンドを有する。図17A-17Bは、オブジェクトグループコピーコマンドを使用して、クライアントオブジェクトと、クライアントオブジェクトにおけるリンクのソースであるオブジェクトの両方をコピーすることにより生じる問題を示している。

20

【0070】

図17Aは、ソース「ledger.xls」オブジェクト1703へのリンク1702を含む「report.doc」オブジェクト1701を示している。リンクは、「ledger.xls」オブジェクトの子孫識別子「7177」及びその区別識別子「2155」を含んでいる。「report.doc」及び「ledger.xls」オブジェクトは、「acme」と称するカンパニーに関連し、「\companies\acme」ディレクトリに含まれる。「\companies\ajax」ディレクトリは、最初、空である。

30

【0071】

「ajax」と称するカンパニーに関連したオブジェクト1701及び1703に対応するオブジェクトを形成するために、ユーザは、オブジェクトグループコピーコマンドを用いて、オブジェクト1701及び1703を「\companies\ajax」ディレクトリへコピーし、コピーされた両方のオブジェクトを編集して、「acme」と称するカンパニーに関連させようとする。

【0072】

図17Bは、オブジェクトグループコピーコマンドを用いて「\companies\acme」ディレクトリから「\companies\ajax」ディレクトリへコピーされたクライアント及びソースオブジェクトを示している。オブジェクト1701及び1703は、図17Aから不変である。「report.doc」オブジェクト1701のコピー1711は、「ledger.xls」オブジェクト1703のコピー1713と同様に、「\companies\ajax」ディレクトリに形成されている。両コピーオブジェクトは、対応するオリジナルオブジェクトと同じ子孫識別子と、対応するオリジナルオブジェクトとは異なる区別識別子とを有し、これらコピーをオリジナルと区別している。「report.doc」オブジェクトのコピー1711は、厳密なコピーであるから、リンク1702の厳密なコピー1712を含んでいる。リンクコピー1712からの矢印で示されたように、リンクコピーは、依然としてオリジナル「ledger.xls」

40

50

s」オブジェクト1703に対して解明するのであって、コピー「ledger.xls」オブジェクト1713に対するものではなく、これは、リンク1712に記憶された区別識別子とは異なる区別識別子を有する。これは殆ど問題ではないが、オリジナル「ledger.xls」オブジェクト1703及びコピー「ledger.xls」オブジェクト1713は同一のままであり、ユーザは、コピー「ledger.xls」オブジェクト1713を直ちに編集して、リンク1712によりコピー「report.doc」オブジェクト1711に組み込まれたデータを変更することができる。リンク1712はオリジナル「ledger.xls」オブジェクト1703に対して解明するので、リンク1712によりコピー「report.doc」オブジェクト1711に組み込まれたデータは変化せず、従って、ユーザの期待を裏切ることになる。

10

【0073】

図17Cは、ファイルのグループをコピーするときに遭遇するリンク解明問題をファシリティがいかに克服するかを示している。ファイルのグループをコピーするときに遭遇するリンク解明問題を克服するために、ファシリティは、図17Cに示すように、コピー「ledger.xls」オブジェクト1713を参照するようにリンク1712を更新する。これは、コピーされるリンクにおけるデータを組み立てながら、グループコピーコマンドを実行し、そしてグループコピーコマンドの完了直後に組み立てられたデータに基づいて必要なリンクを更新することを含む。グループコピーコマンドを実行する間に、ファシリティは2つのテーブルを構成する。クライアントオブジェクトテーブルは、グループコピーコマンドによりコピーされる各クライアントオブジェクトに対するエントリーを含み、これは、クライアントオブジェクトのファイルネームと、クライアントオブジェクトにより含まれた各リンクのオブジェクト識別子とを指示する。ソースオブジェクトテーブルは、グループコピーコマンドによりコピーされる各ソースオブジェクトに対するエントリーを含み、これは、オリジナルソースオブジェクトのオブジェクト識別子と、コピーソースオブジェクトのオブジェクト識別子（これは、オリジナルソースオブジェクトと同じ子孫識別子を有するが、オリジナルソースオブジェクトとは異なる区別識別子を有する）とを指示する。グループコピーコマンドが完了する直後に、ファシリティは、クライアントオブジェクトテーブルを横切る。クライアントオブジェクトテーブルの各エントリーに対し、ファシリティは、各含まれたリンクのオブジェクト識別子を検索する。ファシリティは、ソースオブジェクトテーブルにおいてこのオブジェクト識別子をルックアップし、このオブジェクト識別子を有するリンクのソースがオブジェクトグループコピーコマンドによりコピーされたものであるかどうかを決定する。オブジェクト識別子がソースオブジェクトテーブルにおいてオリジナルソースオブジェクトのオブジェクト識別子として現れる場合には、リンクのソースはオブジェクトグループコピーコマンドによってコピーされており、ファシリティは、クライアントファイルにアクセスするためのクライアントオブジェクトテーブルからのファイルネームを使用して、コピークライアントオブジェクトにおけるリンクのこのオブジェクト識別子をソースオブジェクトテーブルからのコピーオブジェクトのオブジェクト識別子に対して更新する。

20

30

【0074】

ファシリティがリンクを解明するときに、オブジェクト識別子がリンクのオブジェクト識別子に一致するようなオブジェクトを見つけた場合には、ファシリティは、そのオブジェクトに対して直ちに解明する。というのは、所与のオブジェクト識別子をもつオブジェクトは1つしか存在し得ないからである。一方、オブジェクト識別子がリンクのオブジェクト識別子に一致するようなオブジェクトをファシリティが見つけられない場合は、ファシリティは、子孫識別子がリンクの子孫識別子に一致するような多数の候補オブジェクトを見つけることができる。

40

【0075】

それらの子孫識別子がリンクの子孫識別子と一致するので、全ての候補オブジェクトはオリジナルソースオブジェクトに関連され、ある方法を用いて、それらの1つを選択してリンクを解明しなければならない。MatchObjectサブルーチンのフローチャー

50

トは、サブルーチンがこのようなオブジェクトのリストを表示することを示し、ファシリティがリンクを解明すべきところのそれらオブジェクトの1つをユーザが選択できるようにする。ある状況においては、ファシリティは、候補オブジェクト及びクライアントオブジェクトに関する情報を用いて1つの候補オブジェクトを自動的に選択するのが好ましい。例えば、1つの候補オブジェクトのみがクライアントオブジェクトと同じ所有者を有する場合には、ファシリティは、その候補オブジェクトを選択する。同様に、1つの候補オブジェクトがファイルシステムにおいて他の候補オブジェクトよりもクライアントオブジェクトに相当に接近しており、即ち他の候補オブジェクトよりも僅かなディレクトリを経て移動することによりクライアントオブジェクトを含むディレクトリから到達できる場合には、ファシリティは、その接近したオブジェクトを選択する。又、1つの候補オブジェクトのパスネームのファイルネーム部分が、リンクに記憶されたパスネームのファイルネーム部分に一致する場合には、ファシリティは、このオブジェクトを選択する。

10

【0076】

コンピュータシステムの第1ボリュームに記憶されたオブジェクトは、同じコンピュータシステム又は異なるコンピュータシステムの第2ボリュームへ移動することができるので、リンクを解明するときには、ファシリティは、リンクを含むオブジェクトが存在するボリューム以外のボリュームにおいてソースオブジェクトをサーチできるのが好ましい。各々のアクセス可能なボリュームの全処理サーチがおそらく最も有効であるが、アクセス可能なボリュームの番号は、100番代で番号付けされ、全処理サーチを実行不能又は少なくとも不都合なものにする。むしろ、ファシリティは、ソースオブジェクトを含むおそ

20

【0077】

図18は、リンクを解明するためのボリュームサーチの進行を示すフローチャートである。この図は、種々の関連するボリュームが好ましくサーチされる順序を示している。シークしたソースオブジェクトがある段階において見つかった場合には、サーチプログラムが終了となる。

【0078】

リンクは、ソースオブジェクトが初期に位置していた1つ以上のパスネームを含むのが好ましい。リンクに記憶されるパスネームは、クライアントオブジェクトに対するソースオブジェクトの位置を記述する相対的なパスネームと、全ボリュームハイアラキに対するソースオブジェクトの位置を記述する絶対的なパスネームとの両方を含むのが好ましい。ステップ1801において、ファシリティは、これらパスネームのいずれかにソースオブジェクトが依然として存在するかどうかを調べるべくチェックする。ステップ1801は、全ボリュームのサーチを伴わない唯一のサーチ段階であるから、他のサーチ段階の前に行われるのが好ましい。リンクに記憶されたパスネームを用いてステップ1801においてファシリティにより見つかったオブジェクトは、それらのオブジェクト識別子をリンクに記憶されたものと比較することにより照合されるのが好ましい。

30

【0079】

又、リンクは、ソースオブジェクトを以前に保持したボリュームである暗示ボリュームの指示も含むのが好ましい。リンクにおけるこの暗示ボリュームは、リンクが解明されるたびに更新されるのが好ましい。ステップ1802において、ファシリティは、ソースオブジェクトに対して暗示ボリュームをサーチする。

40

【0080】

ステップ1803において、ファシリティは、全てのローカルボリューム、即ちコンピュータシステムに含まれた記憶ボリュームであってクライアントオブジェクトが存在する全ての記憶ボリュームをサーチする。

【0081】

各ユーザは、彼又は彼女が通常ソースオブジェクトを記憶するボリュームのリスト（手動ボリュームリストと称する）を維持するのが好ましい。ステップ1804では、ファシリティは、ソースオブジェクトに対し手動ボリュームにおいてボリュームをサーチする。

50

【0082】

コンピュータシステムは、そのネットワーク接続を用いて別のコンピュータシステムのボリュームへ接続されるたびに、接続ログに入力を行うのが好ましい。

【0083】

ファシリティは、接続ログを、「レジストリ」と称するソースオブジェクトを含むおそれのある他のコンピュータシステムのボリュームの自動ボリュームリストに対するソースとして使用する。ファシリティは、接続ログのエントリがレジストリに追加されるようにレジストリを維持するのが好ましい。ソースオブジェクトが対応するリモートボリュームにおいて普通に見つかる場合にはレジストリにエントリが保持され、そしてソースオブジェクトが対応するリモートボリュームにおいて滅多に見つからない場合にはエントリが破 10
棄される。ステップ1805において、ファシリティは、レジストリにおけるボリュームをサーチする。ファシリティは、ソースオブジェクトを含むおそれのある多数のボリュームをサーチできるので、ソースオブジェクトを見つける機会は、サーチを行うために必要とされるリソースに対して最適なものとすることができる。

【0084】

図19は、例示的なレジストリを示す図である。このレジストリは、一定数のエントリ、ここでは、9つのエントリ1901-1909を含んでいる。各エントリは、ソースオブジェクトを含むリモートボリュームへのパス（「その指示」としても知られている）を含む。エントリ1901は、ソースオブジェクトを最も含み易いボリュームへのパスを含み、エントリ1902は、2番目に含み易いボリュームへのパスを含み、等々となる。ス 20
テップ1805が実行されるときには、ファシリティは、エントリ1901で始めて、レジストリの各エントリごとにボリュームのソースオブジェクトをサーチする。リモートボリュームの1つにおいてソースオブジェクトが見つかった場合には、そのボリュームのエントリは、レジストリにおいてその現在位置から好ましくはレジストリの上部へと上方に移動され、いかなる変位エントリも1位置だけ下方にシフトする。コンピュータシステムが新たなリモートボリュームに接続されるときには、ファシリティは、新たなボリュームに対しレジストリにエントリが存在するかどうか調べるべくチェックする。皆無である場合には、ファシリティは、新たなボリュームへのパスを含むレジストリ、好ましくはレジ 30
ストリの中央に新たなエントリ1910を挿入する。これは、挿入ポイントより下のエントリを各々1位置だけ下方にシフトし、そして最も下のエントリ1909をレジストリから除去し、即ち「その底部から落下」させる。これは、ソースオブジェクトがそのボリュームに見つかった場合にそれが底部から落下する前に新たなエントリを上部へ昇進させる機会を与えることができる。ソースオブジェクトが規則的に見つかるようリモートボリュームに対応するエントリをレジストリに保持するための他の解決策も代用できることが当業者に明らかであろう。例えば、ファシリティは、対応するリモートボリュームにおいて見つかるソースオブジェクトの個数、対応するリモートボリュームにおいてソースオブジェクトが見つかる割合、又は対応するリモートボリュームにおいてソースオブジェクトが最後に見つかったからの時間に基づいてレジストリのエントリを保持又は破棄することができる。

【0085】

ファシリティは、リモートボリュームリストのリストを維持するのが好ましい。各リモートボリュームリストは、個々のコンピュータシステムに維持され、ソースオブジェクトを含み勝ちなボリュームを列挙する。ステップ1806では、ファシリティは、ソースオブジェクトに対し全てのリモートボリュームリストにおける全てのボリュームをサーチする。又、好ましい実施形態では、ファシリティは、リモートボリュームリストのリストそれ自体をボリュームリストとして用いて、ソースオブジェクトに対しリストされたリモートボリュームリストを含むボリュームをサーチする。

【0086】

ステップ1807では、ファシリティは、ローカルマシンが接続された各マシンへサーチ要求を放送する。サーチ要求は、各接続されたマシンがソースオブジェクトに対し全て 50

のボリュームをサーチし、そしてコンピュータシステムがソースオブジェクトを位置決める場合にローカルマシンに報告を戻すように命令する。ステップ1807は、比較的多量の処理及びネットワーク転送リソースを消費する。このように、ファシリティは、このステップを省略するように構成できるのが好ましい。次いで、ファシリティは復帰となる。

【0087】

図18に示すステップにおいてファシリティによりサーチするように識別されたあるボリュームは、ソースオブジェクトに対するオブジェクト識別子の維持をサポートしないことがある。これらの「ダウンレベル」ボリュームの場合には、オブジェクト識別子に基づいてソースオブジェクトをサーチするのではなく、ファシリティは、その基準日付及び時刻に基づいてソースオブジェクトをサーチするのが好ましい。このため、ソースオブジェクトの位置、日付及び時刻がリンクに記憶される。形成日付及び時刻は、詳細な子孫識別子よりも弁別性が悪く、オリジナルソースオブジェクトをコピーから区別するための区別識別子に対応する情報を含まないが、それらは、通常はそれらのオブジェクトがコピーされるときにコピーされるので、子孫の指示を与える。

【0088】

一般にオブジェクトは成形から削除まで同じ子孫識別子を維持するが、既存のオブジェクトに新たな子孫識別子を指定しなければならない場合もある。ユーザは、既存の完全なオブジェクトを新たなオブジェクトに対するテンプレートとしてしばしば使用する。即ち、ユーザは、既存の完全なオブジェクトをロードしたり、それを編集したり、その構造及びフォーマットのほとんどを残しながらその内容のほとんどを削除したり、それを新たなネームのもとでセーブしたりすることがある。これは、一般に、ファシリティによりオブジェクトの単なるコピーとみなされ、ファシリティは、これに応答して、同じ子孫識別子及び新たな区別識別子を指定する。しかしながら、これらの「テンプレート状」オブジェクトは、通常、それらの「テンプレート」とは完全に異なる内容を有し、従って、「テンプレート」オブジェクトに対して形成されたリンクは、「テンプレート状」オブジェクトに対して解明されてはならない。それ故、この状況においては、ファシリティは、新たな子孫識別子及び新たな区別識別子を「テンプレート状」オブジェクトに指定する。「テンプレート状」オブジェクトは、その「テンプレート」オブジェクトの実質的な代替物であると言える。

【0089】

ファシリティは、コピーされたと思われるオブジェクトが実際にテンプレート状であったかどうか決定するために発見的手順を用いるのが好ましい。オブジェクトがロードされたときには、ファシリティは、そのサイズを記録する。オブジェクトが新たなネームでセーブされるときには、ファシリティは、オブジェクトのサイズがそれがロードされてから実質的に減少されたか（例えば、その元のサイズの20%以下）どうか調べるべくチェックする。もしそうであれば、ファシリティは、セーブされるオブジェクトに新たな子孫識別子及び新たな区別識別子を指定する。もしそうでなければ、ファシリティは、セーブされるオブジェクトに、ロードされたオブジェクトと同じ子孫識別子及び異なる区別識別子を指定する。

【図面の簡単な説明】

【0090】

【図1】図1は、リンクの使用を示す図である。

【図2】図2は、リンクを記憶しそして解明する従来の方法を示す図である。

【図3A】図3Aは、コピー及び削除コマンドを用いてソースオブジェクトを再ネーム付けするときに生じる問題を示すブロック図である。

【図3B】図3Bは、コピー及び削除コマンドを用いてソースオブジェクトを再ネーム付けするときに生じる問題を示すブロック図である。

【図3C】図3Cは、コピー及び削除コマンドを用いてソースオブジェクトを再ネーム付けするときに生じる問題を示すブロック図である。

【図4A】図4Aは、コピー及び削除コマンドを用いてソースオブジェクトを移動するときに生じる問題を示すブロック図である。

【図4B】図4Bは、コピー及び削除コマンドを用いてソースオブジェクトを移動するときに生じる問題を示すブロック図である。

【図5】図5は、ロケータテーブルの具現化を示す図である。

【図6】図6は、ファシリティがその中で好ましく動作するところの汎用コンピュータシステムの高レベルブロック図である。

【図7】図7は、オブジェクト識別子がオブジェクトパスネームに対してマップされるオブジェクト識別子テーブルの具現化を示す概略図である。

【図8】図8は、オブジェクト識別子がファイルシステム識別子に対して直接マップされるオブジェクト識別子テーブルの具現化を示す概略図である。 10

【図9】図9は、ソースオブジェクトを位置決めするための第1のMatchObjectサブルーチンを示すフローチャートである。

【図10】図10は、ソースオブジェクトを位置決めするための第2のMatchObjectサブルーチンを示すフローチャートである。

【図11A】図11Aは、いかにファシリティがコピー及び削除問題を克服するかを示すブロック図である。

【図11B】図11Bは、いかにファシリティがコピー及び削除問題を克服するかを示すブロック図である。

【図11C】図11Cは、いかにファシリティがコピー及び削除問題を克服するかを示すブロック図である。 20

【図12A】図12Aは、異なるディレクトリへのリンクを含むオブジェクトの移動の問題をいかにファシリティが克服するかを示すブロック図である。

【図12B】図12Bは、異なるディレクトリへのリンクを含むオブジェクトの移動の問題をいかにファシリティが克服するかを示すブロック図である。

【図13】図13は、多くのアプリケーションプログラムがオブジェクトの新たな改定をセーブするのに使用するプロセスを示すオブジェクト状態タイミング図である。

【図14】図14は、多くのアプリケーションプログラムがオブジェクトの新たな改定をセーブするのに使用するプロセスをいかにファシリティが好ましく取り扱うかを示すオブジェクト状態タイミング図である。 30

【図15】図15は、ProcessDisappearanceサブルーチンのフローチャートである。

【図16】図16は、ProcessAppearanceサブルーチンのフローチャートである。

【図17A】図17Aは、クライアントオブジェクトと、そのクライアントオブジェクトにおけるリンクのソースであるオブジェクトとの両方をコピーするためにオブジェクトグループコピーコマンドを使用することにより生じる問題を示す図である。

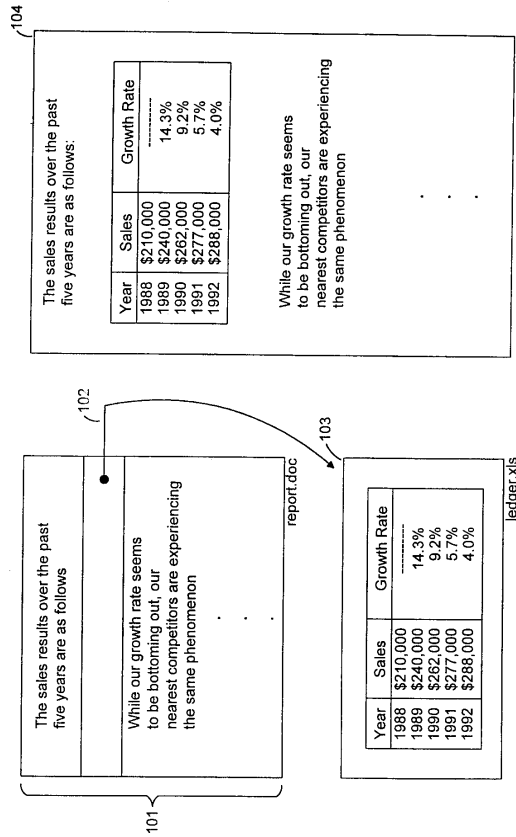
【図17B】図17Bは、クライアントオブジェクトと、そのクライアントオブジェクトにおけるリンクのソースであるオブジェクトとの両方をコピーするためにオブジェクトグループコピーコマンドを使用することにより生じる問題を示す図である。 40

【図17C】図17Cは、ファイルのグループをコピーするときに遭遇するリンク解明問題をファシリティがいかに克服するかを示す図である。

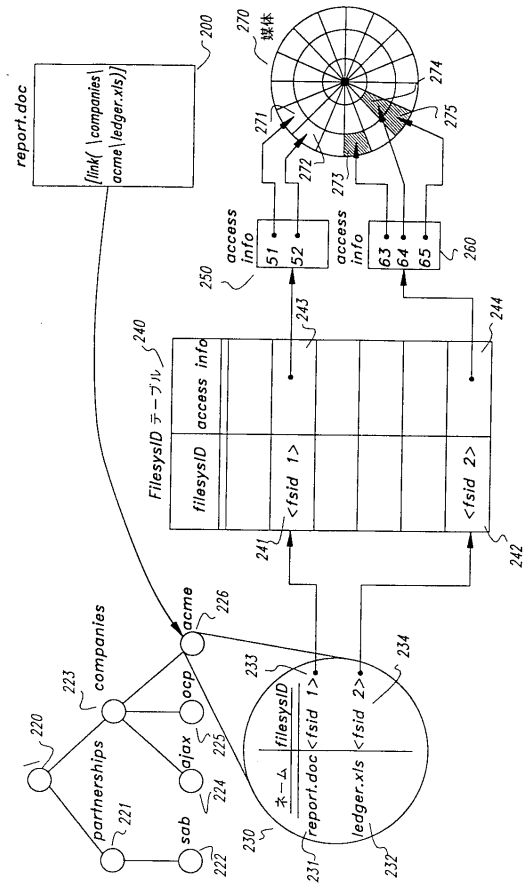
【図18】図18は、リンクを解明するためのボリュームサーチの進行を示すフローチャートである。

【図19】図19は、サンプル登録を示す図である。

【図 1】

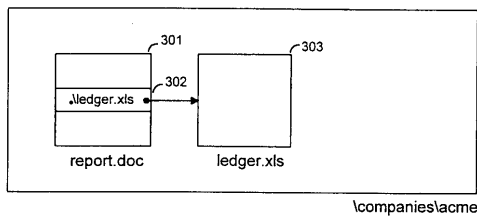


【図 2】

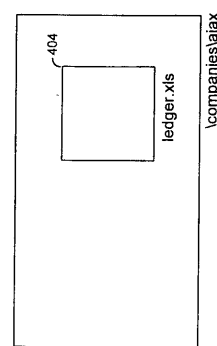


従来例

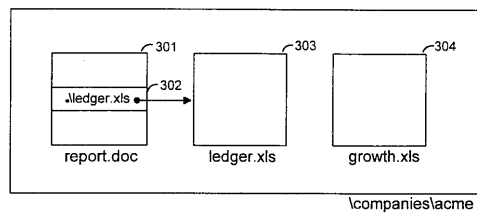
【図 3 A】



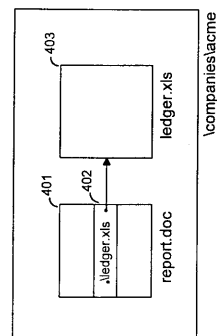
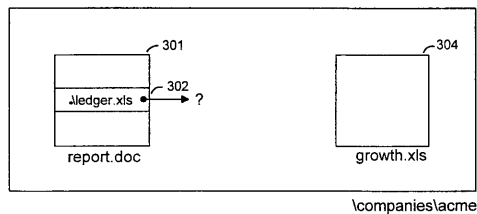
【図 4 A】



【図 3 B】

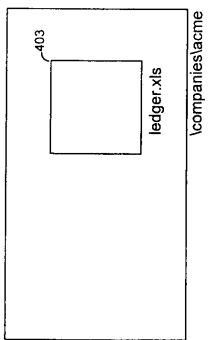
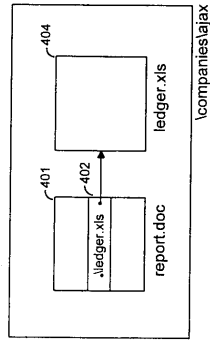


【図 3 C】



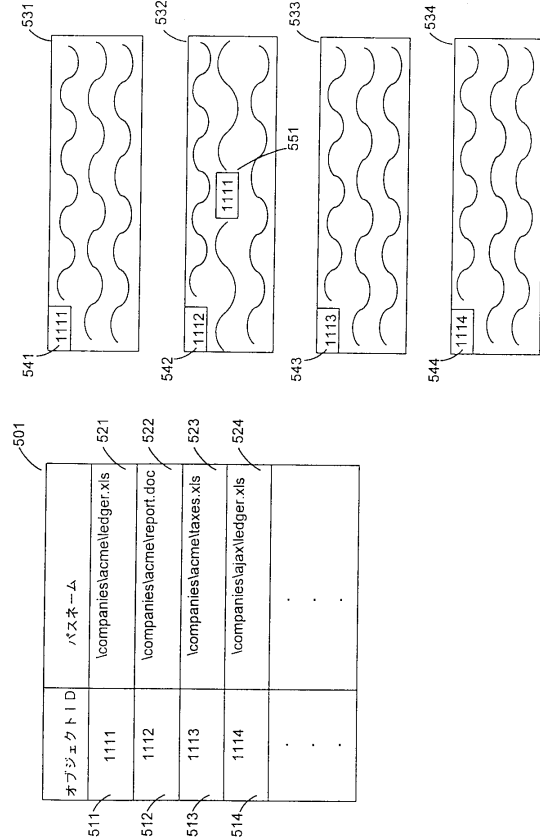
PRIOR ART

【図 4 B】



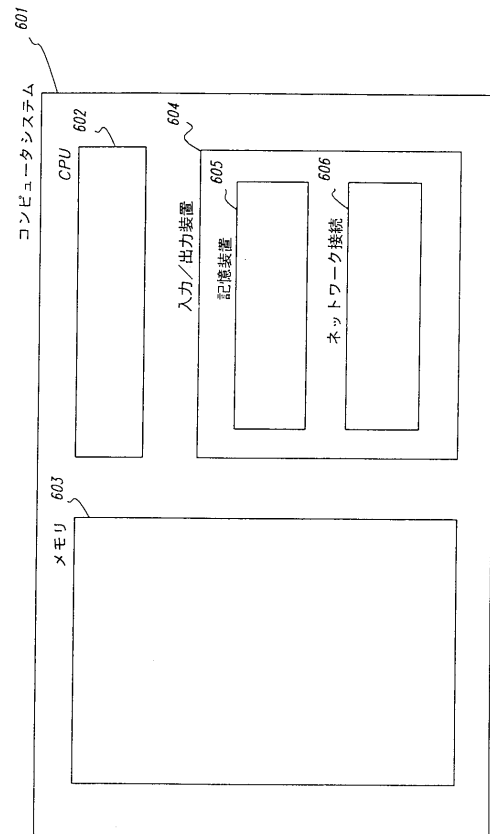
PRIOR ART

【図 5】

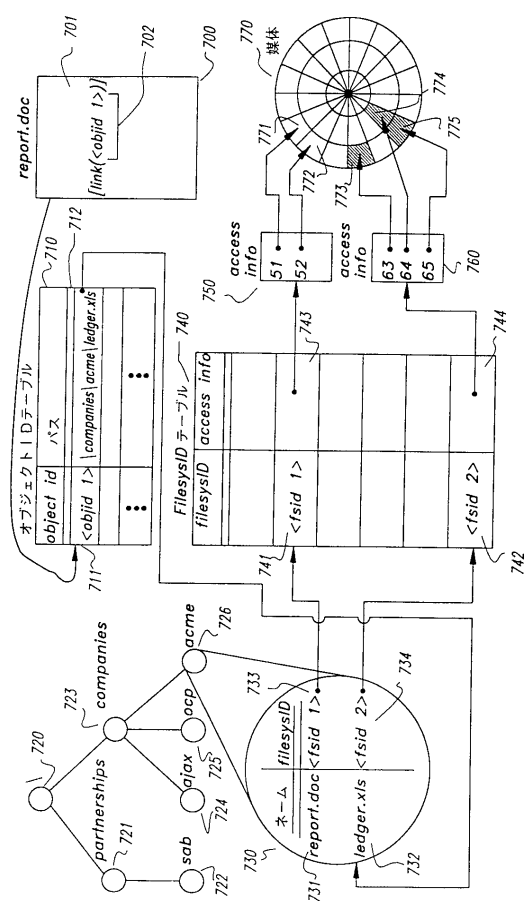


従来例

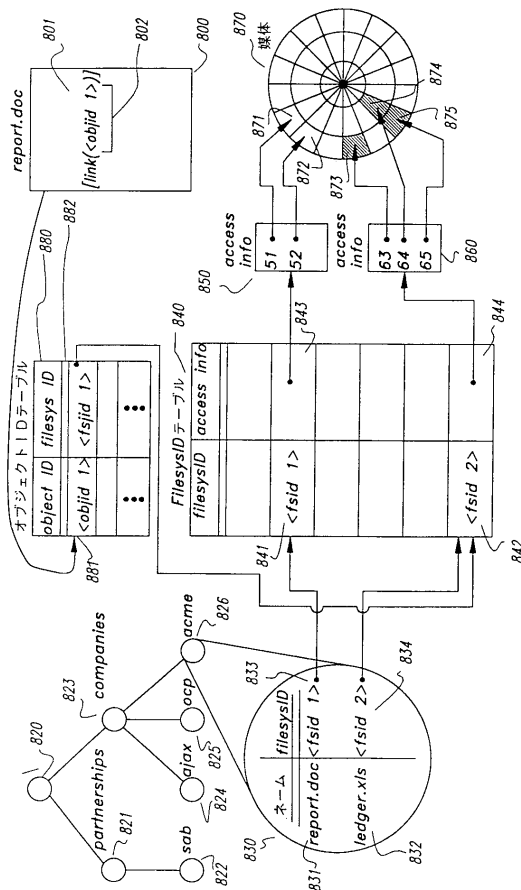
【図 6】



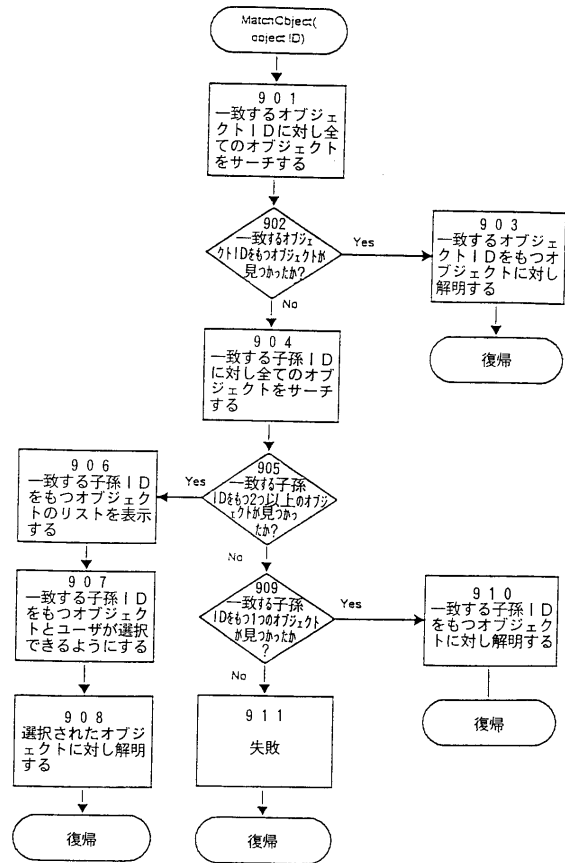
【図 7】



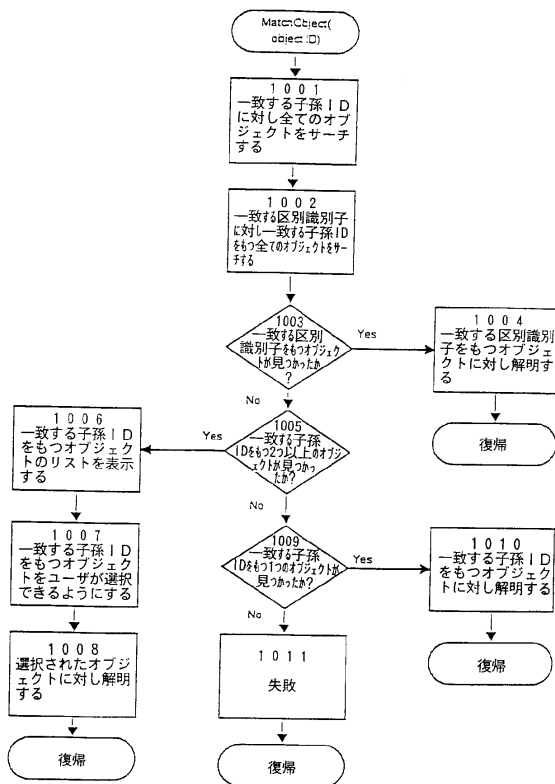
【図 8】



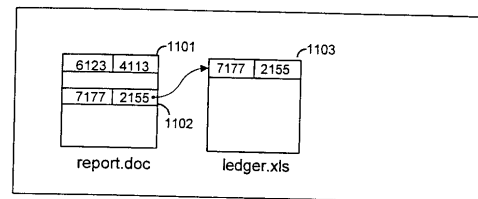
【図 9】



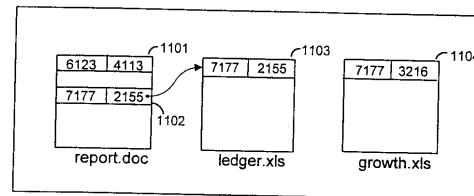
【図 10】



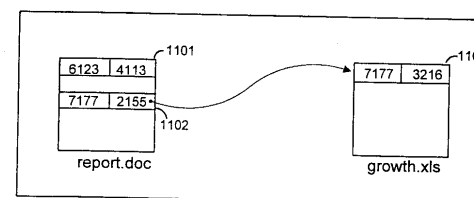
【図 11 A】



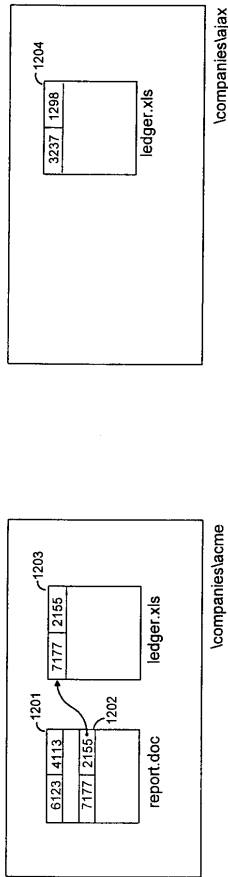
【図 11 B】



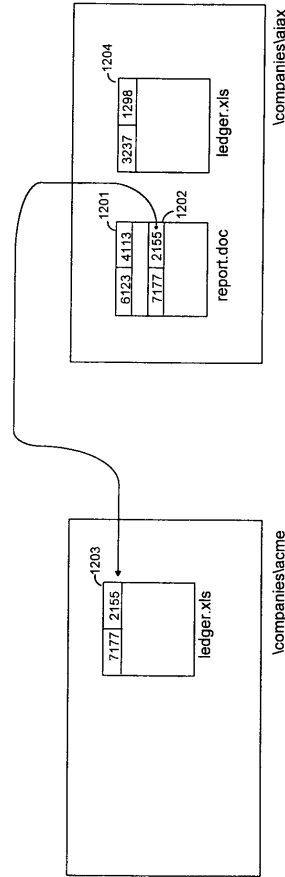
【図 11 C】



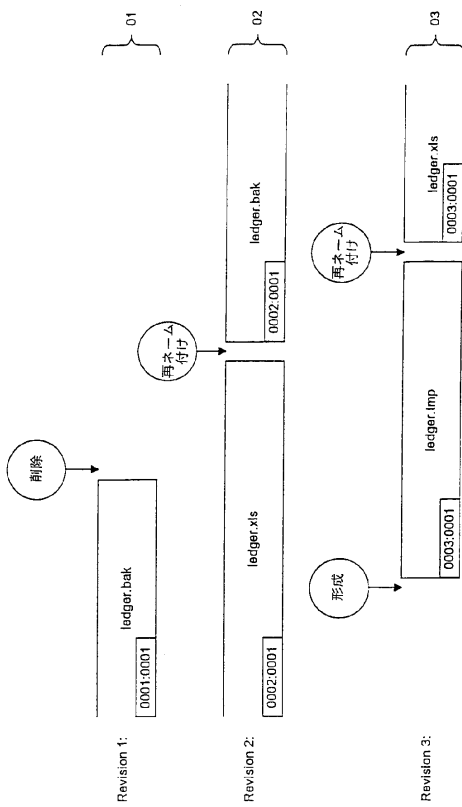
【図 1 2 A】



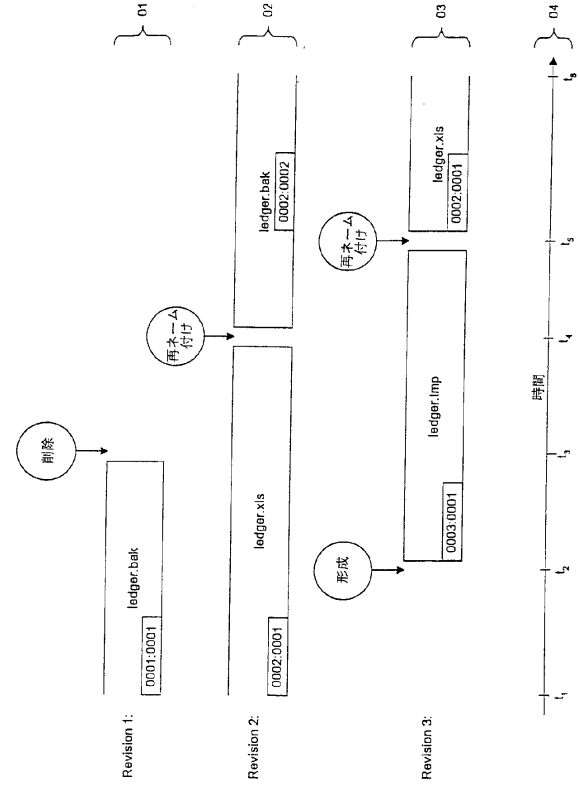
【図 1 2 B】



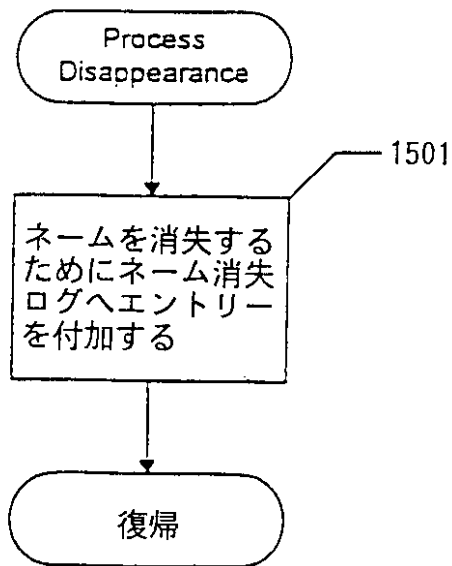
【図 1 3】



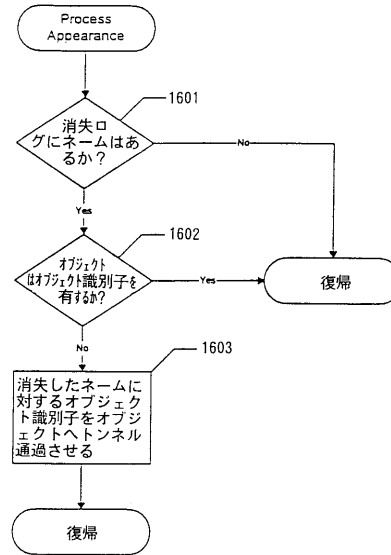
【図 1 4】



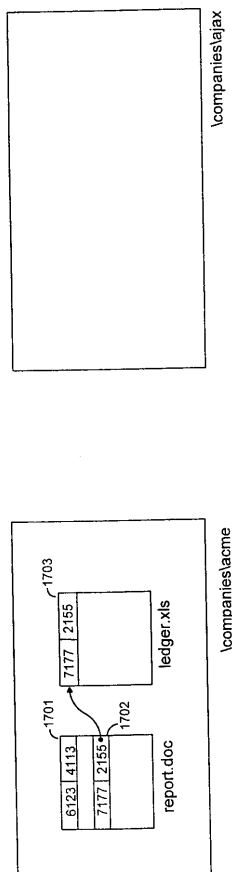
【図 15】



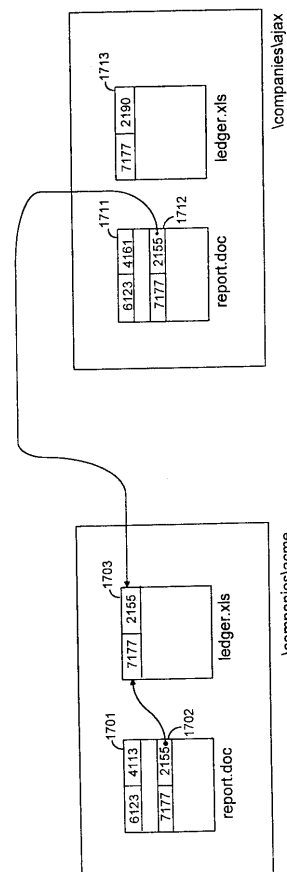
【図 16】



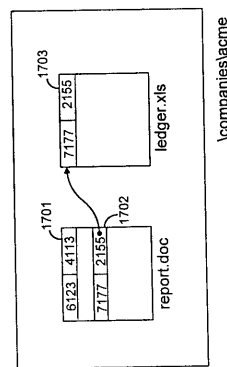
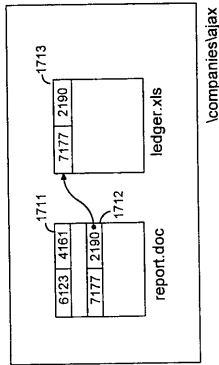
【図 17 A】



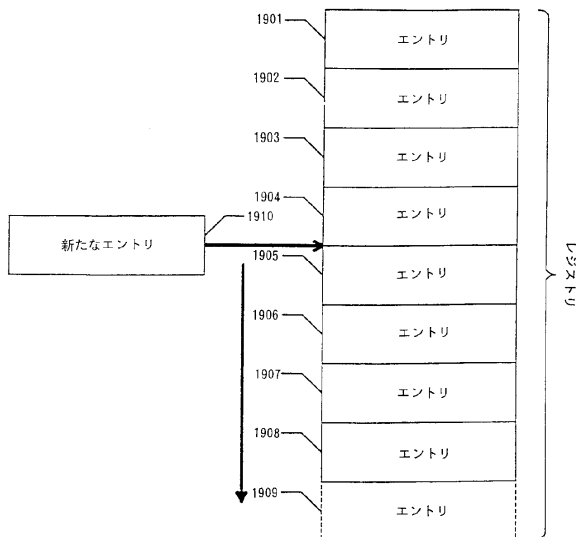
【図 17 B】



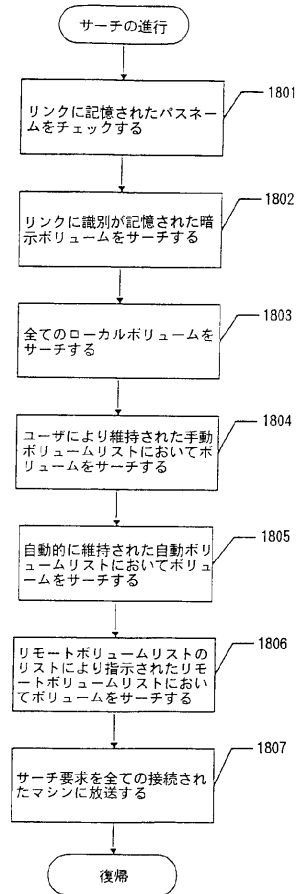
【図 17 C】



【図 19】



【図 18】



フロントページの続き

(74)代理人 100082821

弁理士 村社 厚夫

(74)代理人 100086771

弁理士 西島 孝喜

(72)発明者 モーレル ウィリアム ポール

アメリカ合衆国 ワシントン州 98052 レッドモンド 153 アベニュー ノースイース
ト 7121

(72)発明者 ヤング エドワード クー

アメリカ合衆国 ワシントン州 98133 シアトル ノース ワンハンドレッドアンドセヴン
ティセヴンス ストリート 1829

審査官 桜井 茂行

(56)参考文献 特開平6-290095 (JP, A)

欧州特許出願公開第0304071 (EP, A1)

米国特許第05187786 (US, A)

(58)調査した分野(Int.Cl., DB名)

G06F 12/00

G06F 17/21

NR I サイバーパテント