(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2005-292869
(P2005-292869A)
(43)公開日　平成17年10月20日(2005.10.20)

(51)Int.Cl.[7]	F I		テーマコード（参考）
G06F 12/00	G06F 12/00	531M	5B082
G06F 13/00	G06F 12/00	520E	
	G06F 12/00	545Z	
	G06F 13/00	520D	
	G06F 13/00	625	

審査請求 未請求　請求項の数 7　ＯＬ　（全 29 頁）　最終頁に続く

(21)出願番号　特願2004-102455 (P2004-102455)
(22)出願日　平成16年3月31日 (2004.3.31)

(71)出願人　302064762
株式会社日本総合研究所
東京都千代田区一番町１６番
(74)代理人　100099715
弁理士　吉田　聡
(72)発明者　作山　昭男
東京都千代田区一番町１６番　株式会社日本総合研究所内

Ｆターム(参考)　5B082 DE07　EA01　EA09　EA10　EA11
HA08

(54)【発明の名称】ファイル共有制御システムおよび共有制御プログラム

(57)【要約】

【課題】　ユーザによる煩雑な操作なくメールを利用したファイルのバックアップができ、また、複数のコンピュータ間で、メールを利用してファイルを共有することができる。

【解決手段】　クライアントＰＣ１２は、記憶装置３０のアップロードキューテーブルに記憶された情報およびファイルをメールにて制御サーバ１０に送信する。制御サーバ１０において、センターファイル管理部３４は、情報に含まれるタイムスタンプおよびファイルの内容を示す値に基づいて、ＤＢ３０にファイルを格納し、実ファイル情報テーブルおよび実ファイル世代情報テーブルを更新する。

【選択図】　図１

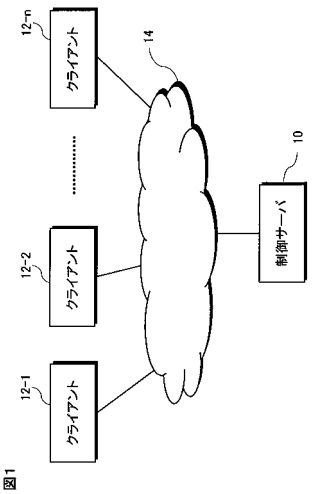


図1

【特許請求の範囲】
【請求項１】
自己が所有するファイルが格納されたディレクトリパス、所有ディレクトリ名を含むレコードを格納する所有ディレクトリ情報テーブル、
メールの添付ファイルとして送信すべきファイルの所有者を示す所有ノード名、所有ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値を含むレコードを格納するアップロードキュー情報テーブル、
記憶装置に記憶されたファイルを管理するファイル管理手段であって、ファイルの作成、変更を検知し、作成、変更されたファイルに関するレコードを、前記アップロードキュー情報テーブルに格納するローカルファイル管理手段、並びに、
前記アップロードキューテーブルのレコードに基づき、前記所有ノード名、ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値とともに、前記ファイルを添付ファイルとしたメールを送信するローカルメール処理手段を有するクライアントコンピュータと、
前記ローカルメール処理手段から送信されたメールを受信するセンターメール処理手段、
前記ネットワークを介して、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、および、所有ディレクトリ名を含むレコードを格納する所有者情報テーブル、
前記所有ノード名、所有ディレクトリ名、ディレクトリパス、および、ファイル名を含むレコードを格納する実ファイル情報テーブル、
ファイル名、ファイルのタイムスタンプ、ファイルの内容を示す情報を含むレコードを格納する実ファイル世代情報テーブル、並びに、
前記センターメール処理手段により受理されたメールに添付されたファイルをデータベースに格納するとともに、実ファイル情報テーブル、および、実ファイル世代情報テーブルの該当レコードを更新するセンターファイル管理手段を有する制御サーバとを備えたことを特徴とするファイル共有制御システム。
【請求項２】
さらに、クライアントコンピュータが、利用したいファイルを所有する他のクライアントコンピュータの利用ノード名、前記利用したいファイルの所有ノード名、利用ディレクトリ名を含むレコードを有する利用ディレクトリ情報テーブルを有し、
前記制御サーバが、前記ローカルメール処理手段から送信され、前記センターメール処理手段により受信されたメールに基づく、利用したいファイルの所有者のノード名、利用ディレクトリ名を含むレコードを格納する利用者情報テーブルを有し、
前記ファイル管理手段が、前記ローカルメール手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、利用ディレクトリ名にしたがって、前記メールに添付されたファイルをデータベースに格納するとともに、前記実ファイル情報テーブル、および、前記実ファイル世代情報テーブルを更新し、
前記センターメール処理手段が、前記所有ノード名、前記利用ディレクトリ情報テーブル中、前記所有ノード名および利用ディレクトリ名を含むレコードの他の利用ノード名により特定されるクライアントコンピュータに、前記ファイルを添付したメールを送信することを特徴とする請求項１に記載のファイル共有制御システム。
【請求項３】
前記所有ディレクトリテーブルのレコードが、ファイルの公開の可否を示す公開フラグ、公開が可能である場合のファイルの更新の可否を示す公開タイプ情報を含み、
前記所有者情報テーブルが、前記公開フラグおよび公開タイプ情報を含み、
前記センターファイル管理手段が、前記所有者情報テーブルの、公開フラグおよび公開タイプ情報を参照して、前記ファイルのＤＢへの格納の可否を判断することを特徴とする請求項２に記載のファイル共有制御システム。
【請求項４】

前記ファイルの内容を示す情報が、ファイルのハッシュ値であり、前記ファイル管理手段が、ハッシュ値を比較することにより、ファイルの内容の一致を判断することを特徴とする請求項１ないし３の何れか一項に記載のファイル共有制御システム。
【請求項５】
さらに、クライアントコンピュータが、所定のタイミングで、所有ノード名、所有ディレクトリ名、ファイル名、ファイルの内容を示す値、タイムスタンプを含む情報を収集する同期処理手段を有し、
前記ローカルメール処理手段が、前記同期処理手段が収集した情報を含むメールを、前記制御サーバに送信し、
前記制御サーバは、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づき、前記ファイルを添付したメールのクライアントコンピュータへの送信、或いは、前記クライアントコンピュータに対する、前記ファイルを添付したメールの送信要求の必要性を判断するセンター同期処理手段を有することを特徴とする請求項１ないし４の何れか一項に記載のファイル共有制御システム。
【請求項６】
請求項１ないし５の何れか一項に記載のファイル共有制御システムにおいて、
自己が所有するファイルが格納されたディレクトリパス、所有ディレクトリ名を含むレコードを格納する所有ディレクトリ情報テーブル、並びに、
メールの添付ファイルとして送信すべきファイルの所有者を示す所有ノード名、所有ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値を含むレコードを格納するアップロードキュー情報テーブルを備えたクライアントコンピュータにより読み出し可能なファイル共有制御プログラムであって、前記クライアントコンピュータを、
記憶装置に記憶されたファイルを管理するファイル管理手段であって、ファイルの作成、変更を検知し、作成、変更されたファイルに関するレコードを、前記アップロードキュー情報テーブルに格納するローカルファイル管理手段、並びに、
前記アップロードキューテーブルのレコードに基づき、前記所有ノード名、ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値とともに、前記ファイルを添付ファイルとしたメールを送信するローカルメール処理手段として機能させることを特徴とするファイル共有制御プログラム。
【請求項７】
請求項１ないし５の何れか一項に記載のファイル共有制御システムにおいて、
前記ネットワークを介して、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、および、所有ディレクトリ名を含むレコードを格納する所有者情報テーブル、
前記所有ノード名、所有ディレクトリ名、ディレクトリパス、および、ファイル名を含むレコードを格納する実ファイル情報テーブル、並びに、
ファイル名、ファイルのタイムスタンプ、ファイルの内容を示す情報を含むレコードを格納する実ファイル世代情報テーブルを有する制御サーバにより読み出し可能なファイル共有制御プログラムであって、
前記制御サーバを、前記ローカルメール処理手段から送信されたメールを受信するセンターメール処理手段、並びに、
前記センターメール処理手段により受理されたメールに添付されたファイルをデータベースに格納するとともに、実ファイル情報テーブル、および、実ファイル世代情報テーブルの該当レコードを更新するセンターファイル管理手段として機能させることを特徴とするファイル共有制御プログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、電子メールを用いたファイルのバックアップおよびその共有を制御するシス

テム、共有制御サーバおよびプログラムに関する。
【背景技術】
【０００２】
近年のコンピュータ機器の性能向上によりパーソナルコンピュータの小型化が進み、
オフィス内で使用されるパーソナルコンピュータが小型化しつつある。また、オフィススペースの制約により、使用されるパーソナルコンピュータも、デスクトップ型のものからノート型の物にシフトしつつある。さらにまた、当然のように、ローカルエリアネットワーク（ＬＡＮ）によってネットワーク接続され、仕事の成果物である各種ファイル（ドキュメントファイルや表計算ソフトのデータファイル等）は、ファイルサーバーに格納するという業務運用が一般化してきている。
【特許文献１】特開２００３－２９６１７５号公報
【発明の開示】
【発明が解決しようとする課題】
【０００３】
しかしながら、ノート型のパーソナルコンピュータが利用されるようになってくると、つぎのような事態が多々発生するようになってきた。
（１）オフィス内のＬＡＮ接続できない場所での作業時にノート型パーソナルコンピュータ（以下、「ノートＰＣ」とも称する。）をＬＡＮから切り離して移動し作業する。
（２）オフィス内にとどまらず、出先にもノートＰＣを持ち出して作業する。
【０００４】
このような利用方法を取る場合、以下のような作業方法をとるケースが頻繁に発生している。
【０００５】
タイプ１
業務に使用する各種のファイルはファイルサーバーから一旦ノートＰＣのローカルディスクにコピーして使用する。
作業終了後、ローカルディスクからファイルサーバーに書き戻す
【０００６】
タイプ２
新規に作成するファイルは、ローカルディスク内にて作成し、作業完了後ファイルサーバーに書き出す。
【０００７】
しかしながら、このような使用方法では、以下のようなトラブルが生じたときに、業務に必要とするファイルが永遠に失われてしまう事態となりかねない。
（１）使用しているノートＰＣが故障した場合
（２）誤操作により必要ファイルを削除してしまった場合
（３）ノートＰＣを紛失（盗難）した場合
上述した状況が発生する背景として、主として以下のような原因があると言われている。
（１）ファイルサーバーへのコピー作業が人間の手作業に依存している。
（２）ノートＰＣのローカルディスクのバックアップ手段になかなかよい物がない。
【０００８】
また、コストの問題と、使用場所を選ばなくても作業できるノートＰＣという特性上、
各ノートＰＣにバックアップ装置をつけるというのも現実的に無理という問題点もある。
【０００９】
その結果、一旦ノートＰＣに障害が発生した場合には、ファイルを復旧できないという致命的な事態となる確率が大きくなってきている。そして、コンピュータ性能の向上によりより大きなローカルディスクが標準的に装着されるようになってきており、その被害影響度合いはますます深刻度を増している。

【００１０】
また、システム設計などにおいて、１つのプロジェクトを複数人で実現することが一般化している。この場合、自己の担当箇所について、パーソナルコンピュータにて作業し、その成果物である各種ファイルを、サーバにも格納し、他の者がこれを閲覧し、場合によっては、更新できるようにしておく。
【００１１】
たとえば、特許文献１には、プロセス定義情報により、グループ名の羅列による処理経路が表わされ、あるグループが処理を終了すると、次のグループが処理を開始できるような協同作業支援システムが開示されている。しかしながら、各作業者が自由に作業を進めることで、全体を進行される場合には、上記システムを利用するのは困難である。また、予めプロセス定義情報を作成しておくという手順も必要となる。
【００１２】
本発明は、ユーザによる煩雑な操作なくメールを利用したファイルのバックアップができ、また、煩雑な操作なく、複数のコンピュータ間で、メールを利用してファイルを共有することができるシステムを提供することを目的とする。
【課題を解決するための手段】
【００１３】
本発明の目的は、自己が所有するファイルが格納されたディレクトリパス、所有ディレクトリ名を含むレコードを格納する所有ディレクトリ情報テーブル、メールの添付ファイルとして送信すべきファイルの所有者を示す所有ノード名、所有ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値を含むレコードを格納するアップロードキュー情報テーブル、記憶装置に記憶されたファイルを管理するファイル管理手段であって、ファイルの作成、変更を検知し、作成、変更されたファイルに関するレコードを、前記アップロードキュー情報テーブルに格納するローカルファイル管理手段、および、前記アップロードキューテーブルのレコードに基づき、前記所有ノード名、ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値とともに、前記ファイルを添付ファイルとしたメールを送信するローカルメール処理手段を有するクライアントコンピュータと、前記ローカルメール処理手段から送信されたメールを受信するセンターメール処理手段、前記ネットワークを介して、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、および、所有ディレクトリ名を含むレコードを格納する所有者情報テーブル、前記所有ノード名、所有ディレクトリ名、ディレクトリパス、および、ファイル名を含むレコードを格納する実ファイル情報テーブル、ファイル名、ファイルのタイムスタンプ、ファイルの内容を示す情報を含むレコードを格納する実ファイル世代情報テーブル、並びに、前記センターメール処理手段により受理されたメールに添付されたファイルをデータベースに格納するとともに、実ファイル情報テーブル、および、実ファイル世代情報テーブルの該当レコードを更新するセンターファイル管理手段を有する制御サーバとを備えたことを特徴とするファイル共有制御システムにより達成される。
【００１４】
本発明によれば、ローカルファイル管理手段により検知され、作成され、更新されたファイルおよび当該ファイルに関する情報が、メールにて制御サーバに送られ、制御サーバに保管される。これにより、ユーザが煩雑な操作をすることなく、ファイルのバックアップ（制御サーバへのファイルのアップロード）や、バックアップされたファイルの取得（制御サーバからのファイルのダウンロード）を実現することができる。
【００１５】
好ましい実施態様においては、さらに、クライアントコンピュータが、利用したいファイルを所有する他のクライアントコンピュータの利用ノード名、前記利用したいファイルの所有ノード名、利用ディレクトリ名を含むレコードを有する利用ディレクトリ情報テーブルを有し、前記制御サーバが、前記ローカルメール処理手段から送信され、前記センターメール処理手段により受信されたメールに基づく、利用したいファイルの所有者のノー

ド名、利用ディレクトリ名を含むレコードを格納する利用者情報テーブルを有し、前記ファイル管理手段が、前記ローカルメール手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、利用ディレクトリ名にしたがって、前記メールに添付されたファイルをデータベースに格納するとともに、前記実ファイル情報テーブル、および、前記実ファイル世代情報テーブルを更新し、前記センターメール処理手段が、前記所有ノード名、前記利用ディレクトリ情報テーブル中、前記所有ノード名および利用ディレクトリ名を含むレコードの他の利用ノード名により特定されるクライアントコンピュータに、前記ファイルを添付したメールを送信する。

【００１６】

本実施の形態によれば、クライアントコンピュータにおいて、他のクライアントコンピュータが所有するファイル中、利用したいものの情報を、利用ディレクトリ情報テーブルのレコードとして用意しておけば、他のクライアントコンピュータが所有するファイルを、利用者が共有することが可能となる。

【００１７】

より好ましい実施態様においては、前記所有ディレクトリテーブルのレコードが、ファイルの公開の可否を示す公開フラグ、公開が可能である場合のファイルの更新の可否を示す公開タイプ情報を含み、前記所有者情報テーブルが、前記公開フラグおよび公開タイプ情報を含み、前記ファイル管理手段が、前記所有者情報テーブルの、公開フラグおよび公開タイプ情報を参照して、前記ファイルのＤＢへの格納の可否を判断する。

【００１８】

たとえば、ファイルの内容を示す情報は、ファイルのハッシュ値であり、センターファイル管理手段が、ハッシュ値を比較することにより、ファイルの内容の一致を判断する。

【００１９】

別の好ましい実施態様においては、さらに、クライアントコンピュータが、所定のタイミングで、所有ノード名、所有ディレクトリ名、ファイル名、ファイルの内容を示す値、タイムスタンプを含む情報を収集する同期処理手段を有し、前記ローカルメール処理手段が、前記同期処理手段が収集した情報を含むメールを、前記制御サーバに送信し、前記制御サーバは、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づき、前記ファイルを添付したメールのクライアントコンピュータへの送信、或いは、前記クライアントコンピュータに対する、前記ファイルを添付したメールの送信要求の必要性を判断するセンター同期処理手段を有する。

【００２０】

また、本発明の目的は、上述した構成のファイル共有制御システムにおいて、自己が所有するファイルが格納されたディレクトリパス、所有ディレクトリ名を含むレコードを格納する所有ディレクトリ情報テーブル、および、メールの添付ファイルとして送信すべきファイルの所有者を示す所有ノード名、所有ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値を含むレコードを格納するアップロードキュー情報テーブルを備えたクライアントコンピュータにより読み出し可能なファイル共有制御プログラムであって、前記クライアントコンピュータを、記憶装置に記憶されたファイルを管理するファイル管理手段であって、ファイルの作成、変更を検知し、作成、変更されたファイルに関するレコードを、前記アップロードキュー情報テーブルに格納するローカルファイル管理手段、並びに、前記アップロードキューテーブルのレコードに基づき、前記所有ノード名、ディレクトリ名、ファイル名、タイムスタンプ、および、ファイルの内容を示す値とともに、前記ファイルを添付ファイルとしたメールを送信するローカルメール処理手段として機能させることを特徴とするファイル共有制御プログラムによっても達成される。

【００２１】

さらに、本発明の目的は、上記構成のファイル共有制御システムにおいて、前記ネットワークを介して、前記ローカルメール処理手段から送信され、センターメール処理手段により受信されたメールに基づく、所有ノード名、および、所有ディレクトリ名を含むレコ

ードを格納する所有者情報テーブル、前記所有ノード名、所有ディレクトリ名、ディレクトリパス、および、ファイル名を含むレコードを格納する実ファイル情報テーブル、並びに、ファイル名、ファイルのタイムスタンプ、ファイルの内容を示す情報を含むレコードを格納する実ファイル世代情報テーブルを有する制御サーバにより読み出し可能なファイル共有制御プログラムであって、前記制御サーバを、前記ローカルメール処理手段から送信されたメールを受信するセンターメール処理手段、並びに、前記センターメール処理手段により受理されたメールに添付されたファイルをデータベースに格納するとともに、実ファイル情報テーブル、および、実ファイル世代情報テーブルの該当レコードを更新するセンターファイル管理手段として機能させることを特徴とするファイル共有制御プログラムによっても達成される。

【発明の効果】

【００２２】

本発明によれば、ユーザによる煩雑な操作なくファイルのバックアップができ、また、煩雑な操作なく、複数のコンピュータ間で、ファイルを共有することができるシステムを提供することが可能となる。

【発明を実施するための最良の形態】

【００２３】

以下、添付図面を参照して、本発明の実施の形態について説明する。図１は、本発明の実施の形態にかかるファイルバックアップ・共有制御サーバ（以下、「制御サーバ」或いは場合によって「センターコントローラ」と称する。）を含むシステム全体の構成を示すブロックダイヤグラムである。図１に示すように、ファイルバックアップ・共有制御サーバ１０と、複数のクライアント・パーソナルコンピュータ（以下、「クライアントＰＣ」或いは場合によって「ノード」と称する）１２－１、１２－２、・・・、１２－ｎは、ネットワーク１４を介して接続可能となっている。また、ネットワーク１４には、制御サーバ１０におけるファイルバックアップやファイル共有の処理について、クライアントＰＣに通知するメールサーバ１６が接続されていても良い。本実施の形態において、ネットワークとして、ＴＣＰ／ＩＰを利用しているが、これに限定されるものではなく、任意の有線或いは無線のネットワークを利用することができる。

【００２４】

図２（ａ）に示すように、制御サーバ１０は、後述するオンラインノード管理テーブル２００、所有者パス(path)情報テーブル２０１、実ファイル情報テーブル２０２、実ファイル世代情報テーブル２０３、および、利用者情報テーブル２０４を含むデータベース２０を有する。また、制御サーバ１０は、ファイルの同期に関する処理を実行する同期処理部（センター同期判断部）２２と、ファイルの読み出し／書き込みおよび送信／受信を管理するファイル管理部（センターファイル管理部）２４と、クライアントマシン１２との間のメール送受信を制御するメール処理部（センターメール処理部）２６とを有している。同期処理部２２、ファイル管理部２４およびメール処理部２６の機能は、制御サーバ１０のメモリなどの記憶装置に記憶されたエージェントプログラム（以下、単に「エージェント」と称する。）により実現される。

【００２５】

図２（ｂ）に示すように、各クライアントＰＣ１２の記憶装置３０には、ノード基本情報テーブル２１１、所有ディレクトリ情報テーブル２１２、利用ディレクトリ情報テーブル２１３、ローカル情報テーブル２１４およびアップロードキュー情報テーブル２１５が記憶される。また、各クライアントＰＣ１２は、ファイルの同期に関する処理を実行する同期処理部（ローカル同期処理部）３２、ファイルをアップロード或いはダウンロードするファイル管理部（ローカルファイル管理部）３４、制御サーバ１０との間のメール送受信を制御するメール処理部（ローカルメール処理部）３６を備えている。記憶装置３２には、エージェント（図示せず）が記憶されており、同期処理部３２、ファイル管理部３４、および、メール処理部３６の機能は、エージェントにより実現される。

【００２６】

本実施の形態においては、クライアントＰＣ（たとえば、クライアントＰＣ１２－１）が、自己の所有するディレクトリのファイルを、制御サーバ１０にバックアップすることができる。このように、ファイルを所有する側のものを、本明細書において「所有者」とも称する。また、本実施の形態においては、クライアントＰＣ１２－１が、他のクライアントＰＣ（たとえば、クライアントＰＣ１２－２）が所有者であるようなディレクトリのファイルを閲覧し、或いは、作成、変更することも可能である。このように、他のクライアントＰＣが所有するファイルを利用する者（たとえば、クライアントＰＣ１２－２）を、本明細書において、「利用者」とも称する。

【００２７】

図３は、ノード基本情報テーブル２１１のデータ構成を示す図である。図３に示すように、ノード基本情報テーブル２１１は、項目として、「自ノード名」、「ノード識別コード」、「PGP暗号Passフェーズ」、「センター電子メールアドレス」、「ユーザタイプ」、「SMTPサーバ名」、「自動受信用メールサーバ名」、「通知受信用電子メールアドレス」、「ファイル受信用電子メールアドレス」、「ファイル受信用アカウント」、「ファイル受信用パスワード」、「メール自動発信時アドレス」、「センターコントローラＩＤ」、「アップロード／ダウンロード作業ディレクトリ」および「前回SyncCheck（同期チェック）日時」を含む。ノード基本情報テーブルには、クライアントＰＣ（以下、「ノード」とも称する。）１２が、制御サーバ（以下、「センターコントローラ」とも称する。）１０にアクセスし、或いは、メールを送受信する際の基本的な情報が含まれる。

【００２８】

図４は、所有ディレクトリ情報テーブル２１２のデータ構成を示す図である。図４に示すように、所有ディレクトリ情報テーブル２１２のレコードは、項目として、「所有ディレクトリPath（パス）」、「削除フラグ」、「所有ディレクトリ呼称」、「センター通知FLG（フラグ）」、「センター受領FLG（フラグ）」、「最大保存世代数」、「削除ファイル保存期間」、「公開フラグ」、「公開パスワード」、「公開タイプ」、「通知メール受領フラグ」、および、「通知メール受信タイプ」を含む。を含む。

【００２９】

所有ディレクトリ情報テーブルのレコードは、クライアントＰＣ（たとえば、クライアントＰＣ１２－１）が所有する一群のファイルを特定するための情報である。たとえば、「所有ディレクトリPath」および「所有ディレクトリ呼称」により特定されるディレクトリに属するファイル群を特定することができる。「公開フラグ」が「ON」である場合には、そのファイル群は、他のクライアントＰＣ（たとえば、クライアントＰＣ１２－２）と共有することができる。

【００３０】

また、「削除FLG」は、ディレクトリの所有者によってディレクトリが削除された場合に、「ON」となる。

【００３１】

「センター通知FLG」は、所有ディレクトリの通知、つまり、当該レコードの情報を、センターコントローラにメールにて通知しているか否かを示す。また、「センター受領FLG」は、通知に対する登録確認を受領しているか否かを示す。

【００３２】

図５は、利用ディレクトリ情報テーブル２１３のデータ構成を示す図である。図５に示すように、利用ディレクトリ情報テーブル２１３のレコードは、項目として、「所有者ノード名」、「所有者ディレクトリ呼称」、「公開パスワード」、「更新可否FLG」、「ファイル格納先基準ディレクトリPath（パス）」、「センター通知FLG」および「センター受領FLG」を含む。「所有者ノード名」および「所有ディレクトリ呼称」により、クライアントＰＣ（たとえば、クライアントＰＣ１２－１）が、ファイルの共有を希望する他のクライアントＰＣ（たとえば、クライアントＰＣ１２－２）のファイル群が特定される。「公開パスワード」により、そのファイル群の共有が許可される。また、「ファイル格納先基準ディレクトリパス」により、クライアントＰＣ（たとえば、クライアントＰＣ１２

－１）が、他のクライアントＰＣ（たとえば、クライアントＰＣ１２－２）から共有を希望するファイル群をダウンロードした際の、クライアントＰＣ（たとえば、クライアントＰＣ１２－１）における格納場所が特定される。「更新可否FLG」は、そのディレクトリのファイルが更新可であるか更新不可であるかが示される。

【００３３】

「センター通知FLG」は、利用ディレクトリの通知、つまり、当該レコードの情報を、センターコントローラにメールにて通知しているか否かを示す。また、「センター受領FLG」は、通知に対する登録確認を受領しているか否かを示す。

【００３４】

図６（ａ）は、ローカルファイル情報テーブル２１４のデータ構成を示す図である。図６（ａ）に示すように、ローカルファイル情報テーブル２１４のレコードは、項目として、「所有者ノード名」、「所有ディレクトリ呼称」、「ローカル相対Path（パス）」、「ファイル名」、「ファイルハッシュ値」、「タイムスタンプ」、「ファイルサイズ」および「アップロード検知作業中FLG」を含む。

【００３５】

「ローカル相対Path」は、所有ディレクトリや利用ディレクトリとして指定したローカルディスク内のPathを基準とした相対Pathであり、これにより、ファイルの記憶装置３０中の格納位置を特定することができる。「ファイルハッシュ値」は、ファイルのMD5ハッシュ値である。ローカルファイル情報テーブルのレコードは、クライアントＰＣの記憶装置３０（ローカルディスク）に保管されるファイルの情報を示す。この情報を使用して、アップロードすべきファイルが判断される。

【００３６】

図６（ｂ）は、アップロードキュー情報テーブル２１５のデータ構成を示す図である。図６（ｂ）に示すように、アップロードキュー情報テーブル２１５のレコードは、項目として、「アップロード検知日時」、「所有ノード名」、「所有ディレクトリ呼称」、「ファイル名」、「ファイルハッシュ値」、「タイムスタンプ」、「ファイルサイズ」、「削除FLG（フラグ）」、「センターアップロードFLG」および「ファイル別名」を含む。「削除FLG」は、ファイルが記憶装置３０（ローカルディスク）から削除されているか否かを示す。また、「センターアップロードFLG」は、ファイルを制御サーバにアップロード済みか否かを示す。

【００３７】

図７は、オンラインノード管理テーブル２００のデータ構成を示す図である。図７に示すように、オンラインノード管理テーブルのレコードは、項目として、「ノード名」、「仮登録FLG（フラグ）」、「ノード識別コード」、「ユーザタイプ」、「ノード発信電子メールアドレス」、「通知用電子メールアドレス」、「ファイル受信用電子メールアドレス」、「ノード登録日」、「受信頻度タイプ」および「ノード登録申請者名」を含む。このレコードに含まれる項目の値により、制御サーバ１０は、ネットワーク１４を介して接続したクライアントＰＣ１２を特定することができる。

【００３８】

図８は、所有者情報テーブル２０１のデータ構成を示す図である。図８に示すように、所有者パス情報テーブル２１０のレコードは、項目として、「ノード名」、「所有ディレクトリ呼称」、「削除FLG（フラグ）」、「削除日時」、「最大保存世代数」、「削除ファイル保存期間」、「公開フラグ」、「公開パスワード」、「公開タイプ」、「通知メール受領フラグ」および「通知メール受信タイプ」を含む。

【００３９】

このレコードは、クライアントＰＣ１２の所有ディレクトリ情報テーブル２１２のレコードが、制御サーバ１０に登録される際に新規に作成される。したがって、「ノード名」は、クライアントＰＣのノード名に相当し、また、所有者パス情報テーブルのレコード中、「削除FLG」、「所有ディレクトリ呼称」、「最大保存世代数」、「削除ファイル保存期間」、「公開フラグ」、「公開パスワード」、「公開タイプ」、「通知メール受領フラ

グ」および「通知メール受信タイプ」の値が、所有ディレクトリ情報のレコード中、対応する項目の値として格納される。

【００４０】

図９（ａ）は、利用者情報テーブル２０４のデータ構成を示す図である。図９（ａ）に示すように、利用者情報テーブル２０４のレコードは、項目として、「利用ノード名」、「所有ノード名」、「利用ディレクトリ呼称」および「削除フラグ」を有する。これらは、利用者であるクライアントＰＣを特定する情報である。「利用ノード名」は、利用者のクライアントＰＣのノード名に相当し、「所有ノード名」は、利用者が利用するファイルやディレクトリの所有者のノード名を示す。「削除フラグ」は、利用者によってディレクトリが削除されたか否かを示す。

【００４１】

図９（ｂ）は、実ファイル情報テーブル２０２のデータ構成を示す図である。図９（ｂ）に示すように、実ファイル情報テーブル２０２のレコードは、項目として、「所有ノード名」、「所有ディレクトリ呼称」、「相対Path（パス）」、「実ファイル名」、「サーバファイル名」および「削除フラグ」を有する。「所有ノード名」は、クライアントＰＣ１２のノード名に対応し、「所有ディレクトリ呼称」は、所有ディレクトリ情報テーブルのレコード中、「所有ディレクトリ呼称」に対応する。「相対Path」は、ファイルが、クライアントＰＣ１２において、所有ディレクトリ内のサブディレクトリに格納されている場合の、所有ディレクトリからの相対パスを示す。また、「サーバファイル名」は、当該ファイルを制御サーバ１０において格納する場合のファイル名である。この実ファイル情報テーブル２０２のレコードにより、クライアントＰＣ１２中のファイルを特定することができる。

【００４２】

図９（ｃ）は、実ファイル世代情報テーブル２０３のデータ構成を示す図である。図９（ｃ）に示すように、実ファイル世代情報テーブル２０３のレコードは、項目として、「サーバファイル名」、「登録日時」、「登録ディレクトリ」、「登録ノード名」、「格納ファイルサイズ」、「ファイルサイズ」、「ハッシュ値」、「圧縮後ハッシュ値」および「タイムスタンプ」を有する。「登録日時」は、ファイルが制御サーバ１０の記憶装置に格納された日時に相当する。「登録ディレクトリ」は、制御サーバ１０にファイルを格納するときのパスに相当する。制御サーバ１０は、記憶装置に、格納ファイル情報テーブル（図示せず）を記憶しておく。この格納ファイル情報テーブルのレコードには、クライアントＰＣ１２からアップロードされたファイルを、制御サーバ１０にて保存する、記憶装置内のディレクトリパスに相当する「ファイルバックアップ基準Path（パス）」が含まれる。登録ディレクトリは、格納ファイル情報テーブルの何れかのレコードの値と一致する。

【００４３】

次に、本システムを構成するクライアントＰＣを利用するユーザのタイプについて説明する。利用者のネットワーク環境（特に電子メール環境）をから分類すると、以下の２つのタイプにユーザを分けることができる。

【００４４】

タイプ１：ファイル送受信専用アカウント可の環境、つまり、ローカルファイルのアップロード/センターコントローラからのダウンロードは全てバックグラウンドで自動に行われる環境

タイプ２：ファイル送受信専用アカウント不可の環境、つまり、ローカルファイルのアップロードは全てバックグラウンドで自動的に行われるが、ダウンロード用のメールアドレスを利用できない為、通常のメールと同じメールアドレスに送付する環境

また、本システムにおいては、クライアントＰＣの記憶装置のディレクトリ（ローカルディレクトリ：サブディレクトリも含む）のファイルを、センターコントローラ上にディレクトリ呼称として登録し、ネットワーク内に、非公開、公開（更新不可）或いは公開（更新可）として利用する形をとる。この場合に、ディレクトリの所有者と、ディレクトリ

の利用者という２つの形態で利用方法を使い分ける。
【００４５】
ディレクトリ所有者は、そのローカルディレクトリ内にファイルを作成・修正・削除する事が出来る。センターコントローラにディレクトリ呼称として登録したディレクトリ（サブディレクトリも含む）内のファイルに対するその変更（新規・修正・削除）内容は、センターコントローラ側に自動的にアップロードされ、センターコントローラにおいて、履歴管理される。また、そのディレクトリ呼称毎に、以下の定義が可能である。
（１）利用者からは見えない（非公開）：（ローカルディスクのバックアップ）
（２）利用者からは参照のみ可（公開・更新不可）：ファイルの自動配布
（３）利用者が新規・修正・削除できる（公開・更新可）：共同作業によるファイルの作成
また、ディレクトリ利用者は、ネットワーク内に公開されたディレクトリ呼称を使用する場合、利用ディレクトリとして登録することにより、そのディレクトリ呼称の情報を自動的にダウンロード（さらには、ディレクトリが更新可であれば自動的にアップロード）することが出来る。
【００４６】
次に、ファイルの受信タイプについて説明する。本システムでは、クライアントＰＣがファイルを受取る場合に、以下の２つの形態をサポートすることができる。
（１）ファイル受信用の専用メールアカウントが使用できる環境（ユーザタイプ＝１）
（２）通常の電子メールアカウントでファイルを受信できる環境（ユーザタイプ＝２）
ファイル受信用の専用メールアカウントが使用できる環境においては、クライアントＰＣにインストールされたプログラム（メール処理部）が、定期的にメールサーバ（図示せず９に対して受信チェックし、メールが受信していた場合は、クライアントＰＣにダウンロードし、暗号を復号化して所定のディレクトリにファイルを格納する。
【００４７】
その一方、通常の電子メールアカウントでファイルを受信できる環境では、クライアントＰＣ側にダウンロードするために、利用者が普段使用する電子メールソフトを利用する。ファイルが添付されたメールは電子メールソフト上では通常の添付ファイルとして認識されるが、拡張子を特殊なものに設定して、マウスによるダブルクリックなどによって、拡張子対応のプログラムの自動起動機能によりファイルの複号化、所定のディレクトリへの保存が自動で行えるようにする。
【００４８】
この２の環境を使い分ける事により利用者が使用できる環境に合わせたファイル受信が可能となる。
【００４９】
次に、ファイル送信タイプについて説明する。本システムでは、クライアントＰＣ側からファイルを送信する場合、以下の形態をサポートする。
【００５０】
クライアントＰＣ側でファイル送信の必要性（新規/変更/削除ファイルが発生）な時に自動的に、対象ファイルを暗号化してセンターコントローラ宛の電子メールとして送信する。ただし、但し、送信処理は、ネットワークに接続し、SMTPサーバ（図示せず）との通信が可能な場合にのみ実行される。
【００５１】
その結果、以下の２つのユーザをサポートすることが可能である。
（１）ネットワークに常時接続して利用しているユーザ
（２）ネットワークには必要な時のみ接続しているユーザ
上述したように構成されたクライアントＰＣ１２および制御サーバ１０における処理について以下に説明する。
【００５２】
まず、クライアントＰＣ１２から制御サーバ１０への初期的な基本情報の登録処理につ

いて簡単に説明する。

【００５３】

初期作業として、クライアントＰＣ１２のメール処理部３６は、以下の情報を添付ファイルとして、制御サーバ１０に送信する。

【００５４】

自ノード名
ユーザタイプ
PGP暗号Passフェーズ
センター電子メールアドレス
SMTPサーバ名
メールサーバ名
通知受信用電子メールアドレス
ファイル受信用電子メールアドレス
ファイル受信用アカウント
ファイル受信用パスワード
メール自動発信時電子メールアドレス
アップロード/ダウンロード作業ディレクトリPath

ここで、「ユーザタイプ」とは、前述した利用者のネットワーク環境、つまり、

【００５５】

タイプ１：ファイル送受信専用アカウント可の環境、つまり、ローカルファイルのアップロード/センターコントローラからのダウンロードは全てバックグラウンドで自動に行われる環境

【００５６】

タイプ２：ファイル送受信専用アカウント不可の環境、つまり、ローカルファイルのアップロードは全てバックグラウンドで自動的に行われるが、ダウンロード用のメールアドレスを利用できない為、通常のメールと同じメールアドレスに送付する環境

のいずれかである。

【００５７】

上記以外の情報は、基本的に、クライアントＰＣ１２のノード基本情報テーブル２１１に格納しておいたものに相当する。管理サーバ１０は、ネットワーク１４を介して受信した情報を、オンラインノード管理テーブルのレコードなどに格納しておく。

【００５８】

次に、所有ディレクトリの登録および利用ディレクトリの登録について説明する。

【００５９】

ユーザは、クライアントＰＣ１２を操作して、所有ディレクトリ情報テーブル２１２のレコードに値を格納しておく。クライアントＰＣ１２のメール処理部３６は、所有ディレクトリ情報テーブル２１２を読み出して、これを添付ファイルとしたメールを、制御サーバ１０に送信する。制御サーバ１０は、受信したファイルの情報を、所有者情報テーブル２０１のレコードとして格納し、その後、登録完了メールを、クライアントＰＣ１２に送信する。

【００６０】

クライアントＰＣ１２のメール処理部３６は、登録完了メールを受信すると、所有ディレクトリ情報テーブル２１２の該当レコード中、「センター受領FLG」を「ON」にする。

【００６１】

利用ディレクトリの登録についても同様の手順で実現できる。

【００６２】

次に、本実施の形態にかかるクライアントＰＣにおけるファイルのアップロードおよびダウンロードの基本的な流れについて説明する。図１０は、制御サーバ１０をファイルバックアップ装置として利用する例を示す。クライアントＰＣ１２－１が、制御サーバ１０の所有者情報テーブル２０１にその情報を登録したディレクトリのファイルを作成、更新

すると（ステップ１００１）、作成、更新にかかるファイルを添付したメールが制御サーバ１０に送信される（ステップ１００２）。制御サーバ１０は、作成、変更されたファイルを履歴として格納する（ステップ１００３）。これにより、ファイルのアップロードが実現できる。また、クライアントＰＣ１２－１の要求などにしたがって、制御サーバ１０から、ファイルが添付されたメールが送信される（ステップ１００４）。これにより、ファイルのアップロードが実現される。

【００６３】

図１１は、ファイルを共有する例を示す図である。まず、所有者であるクライアントＰＣ１２－１が、制御サーバ１０の所有者情報テーブル２０１に情報を登録しておく。この際に、「公開フラグ＝公開」、「公開タイプ＝更新可」としておく（ステップ１１０１）。また、他のクライアントＰＣ１２－２等は、制御サーバ１０の利用者情報テーブル２０４に、所有者のディレクトリの利用を登録しておく。これにより、他のクライアントＰＣ１２－２等は、そのディレクトリのファイルを更新等することが可能となる（ステップ１１０２）。クライアントＰＣ１２－２によりファイルが更新されると、他のクライアントＰＣ（クライアントＰＣ１２－１、１２－３など）には、更新されたファイルが添付されたメールが送信される（ステップ１１０３）。このようにしてファイルの共有が実現できる。

【００６４】

以下、より具体的に、クライアントＰＣ１２における処理、制御サーバ１０における処理、および、クライアントＰＣ１２と制御サーバ１０との間の処理について説明する。

【００６５】

まず、クライアントＰＣ１２における、制御サーバ１０にアップロードすべきファイルの検知処理について説明する。

【００６６】

図１２に示すように、クライアントＰＣのファイル管理部３４は、一定間隔ごとに、所有ディレクトリ情報のレコードごとに、当該所有ディレクトリPath内の全てのファイルに関して、以下のチェックを実行する。

（１）ローカルファイル情報テーブルのアップロード検知作業中FLGを全てONに変更する（ステップ１２０１）。

（２）所有ディレクトリ情報の所有ディレクトリPath配下（サブディレクトリを含む）の全ファイルについて以下の処理を行う。

【００６７】

Ａ．ローカルファイル情報テーブル内にPath,ファイル名が同じものが存在するかチェックする（ステップ１２０２）。

【００６８】

Ｂ．存在しない場合には（ステップ１２０２でノー(No)）、

ａ）ローカルファイル情報テーブルに，以下のレコードを追加する（ステップ１２０３）。

所有者ノード名＝自ノード名

所有ディレクトリ呼称　　＝所有ディレクトリ情報テーブルの所有ディレクトリ呼称

ファイル名＝該当ファイルのファイル名

ファイルハッシュ値＝該当ファイルのMD5ハッシュ値

タイムスタンプ＝該当ファイルのタイムスタンプ

ファイルサイズ＝該当ファイルのファイルサイズ

ローカルPath＝所有ディレクトリを基準とした相対Path

【００６９】

ｂ）基本情報ファイルテーブルのアップロード/ダウンロード作業ディレクトリに該当ファイルをコピーする（ステップ１２０４）。なお。コピー先のファイル名はユニークに採番されたファイル名とする。

【００７０】
ｃ）アップロードキュー情報テーブルに、以下のレコードを追加する（ステップ１２０５）。
アップロード検知日時＝該当ファイルを見つけたマシン日時
所有ディレクトリ呼称＝所有ディレクトリ情報テーブルの所有ディレクトリ呼称
ファイル名＝該当ファイルのファイル名
ファイルハッシュ値＝該当ファイルのMD5ハッシュ値
タイムスタンプ＝該当ファイルのタイムスタンプ
ファイルサイズ＝該当ファイルのファイルサイズ
削除FLG＝OFF
アップロード通知メールFLG＝OFF
センターアップロードFLG＝OFF
ファイル別名＝コピー後のファイル名
【００７１】
ｄ）ローカルファイル情報の該当レコードのUpload検知作業中FLGをOFFに変更する（ステップ１２０６）。
【００７２】
Ｃ．存在する場合には、
ａ）ローカルファイル情報テーブルに登録されているタイムスタンプと該当ファイルのタイムスタンプをチェックする（ステップ１２０７）。
【００７３】
ｂ）タイムスタンプが同じである場合（ステップ１２０７でイエス(Yes)）
ローカルファイル情報テーブルのアップロード検知作業中FLGをOFFに変更する（ステップ１２０６）。その後、次のファイルのチェックに進む
【００７４】
ｃ）タイムスタンプが異なる場合（ステップ１２０７でノー(No)）
該当ファイルのMD5ハッシュ値を算出する。
【００７５】
ローカルファイル情報のファイルハッシュ値を算出したMD5ハッシュ値を比較する（ステップ１２０８）。
【００７６】
ハッシュ値が同じである場合には（ステップ１２０８でイエス(Yes)）、ローカルファイル情報のアップロード検知作業中FLGをOFFに変更し（ステップ１２０６）、次のファイルのチェックに進む。
【００７７】
その一方、ハッシュ値が異なる場合には、基本情報ファイルテーブルのアップロード/ダウンロード作業ディレクトリに該当ファイルをコピーする（ステップ１２０９）。なお、コピー先のファイル名はユニークに採番されたファイル名とする。
【００７８】
次いで、アップロードキュー情報テーブルに、以下のレコードを追加する（ステップ１２１０）。
アップロード検知日時＝該当ファイルを見つけたマシン日時
所有ディレクトリ呼称＝所有ディレクトリ情報テーブルの所有ディレクトリ呼称
ファイル名＝該当ファイルのファイル名
ファイルハッシュ値＝該当ファイルのMD5ハッシュ値
タイムスタンプ＝該当ファイルのタイムスタンプ
ファイルサイズ＝該当ファイルのファイルサイズ
削除FLG＝OFF
アップロード通知メールFLG＝OFF
センターアップロードFLG＝OFF

ファイル別名＝コピー後のファイル名
【００７９】
この後、ローカルファイル情報テーブルの該当レコードのアップロード検知作業中FLGをOFFに変更する（ステップ１２０６）。
【００８０】
この処理では、基本的には、新たにファイルが作成され、或いは、更新されている場合には、アップロード／ダウンロード作業ディレクトリに、そのファイルがコピーされ、また、ファイルの情報のレコードが、アップロードキューテーブルに追加される。
【００８１】
また、全てのファイルについて処理を実行し終わった状態で、ローカルファイル情報テーブル内にアップロード検知作業中FLGがONのレコードが残っている場合もある（図１３のステップ１３００でイエス(Yes)）。これは、ファイルが削除してしまった場合に相当する。ファイルの削除を考慮して、ファイル管理部３４は、図１３に示す処理を実行する。
【００８２】
Ａ．まず、ファイル管理部３４は、アップロードキュー情報テーブルに、以下のレコードを追加する（ステップ１３０１）。
アップロード検知日時＝マシン日時
所有者ノード名＝自ノード名
所有ディレクトリ呼称＝ローカルファイル情報の所有ディレクトリ呼称
ファイル名＝ローカルファイル情報のファイル名
ファイルハッシュ値＝ローカルファイル情報のMD5ハッシュ値
タイムスタンプ＝ローカルファイル情報のタイムスタンプ
ファイルサイズ＝ローカルファイル情報のファイルサイズ
削除FLG＝ON
アップロード通知メールFLG＝OFF
センターアップロードFLG＝OFF
ファイル別名＝未登録
【００８３】
Ｂ．次いで、ローカルファイル情報テーブルの該当レコードを削除する（ステップ１３０２）。
後述するように、アップロードキュー情報テーブルに、ファイルが削除されたことを示すレコードを格納し、このレコードが制御サーバ１０に伝達されることにより、制御サーバ１０においても、ファイルが削除されたことを把握できる。
【００８４】
利用ディレクトリ情報テーブルのレコード中の、ファイル格納ローカルPath（サブディレクトリも含む）の全てのファイルについて、図１２および図１３に示すものと同等の処理を実行することにより、作成、更新されたファイルや削除されたファイルを検出することができる。
【００８５】
次に、クライアントＰＣ１２から制御サーバ１０へのファイルのアップロード処理について説明する。クライアントＰＣ１２のファイル管理部３４は、一定時間ごとに、アップロードキュー情報テーブルを参照して、レコード中、センターアップロードFLGがOFFであるレコードが存在するか否かを判断する。センターアップロードFLGがOFFであるレコードが存在する場合には、ファイル管理部３４が必要な情報を収集し、メール処理部３６が、収集された情報やファイルを添付したメールを、制御サーバ１０に送信する。
【００８６】
図１４に示すように、まず、アップロードメール用の制御ファイルが生成される（ステップ１４０１）。制御ファイルには以下の情報が含まれる。
【００８７】

情報数：１つのメールに幾つのファイルに関する情報が入っているかを示す（基本的には「１」をセットする）。
メールタイプ：アップロードメールである事をキーワード（例えばUPLOAD）
アップロードノード名：基本情報テーブル中の自ノード名
アップロードノード識別コード：基本情報テーブル中の自ノード識別コード
所有ノード名：ローカルファイル情報テーブル中の該当項目
所有ディレクトリ呼称：ローカルファイル情報テーブル中の該当項目
ファイル名：ローカルファイル情報テーブル中の該当項目
ファイルハッシュ値：ローカルファイル情報テーブル中の該当項目
ファイルタイムスタンプ：ローカルファイル情報テーブル中の該当項目
ファイルサイズ：ローカルファイル情報テーブル中の該当項目
相対Path：ローカルファイル情報テーブル中のローカル相対Path
ファイルハッシュ値：ローカルファイル情報テーブルの該当項目
ファイル別名：アップロードキュー情報テーブル中の該当項目

【００８８】

次いで、アップロードメール用メールファイルが作成される（ステップ１４０２）。ここでは、ステップ１４０１で作成されたアップロードメール用の制御ファイル、および、該当するローカルファイル情報テーブルのファイル別名で特定されるファイルをアーカイブファイルとして圧縮する。ただし、ローカルファイル情報テーブルのレコードにおいて、削除FLG＝ONの場合には、実ファイルはアーカイブされない。

【００８９】

圧縮されたファイルが暗号化される（ステップ１４０３）。暗号化には、たとえば、PGPの鍵方式が利用される。

【００９０】

次いで、メール処理部３６は、生成されたアップロードメールを、ノード基本情報テーブルに記載されたセンター電子メールアドレス宛に送信する（ステップ１４０４）。なお、ここでは、発信者は、ノード基本情報テーブルに記載された通知受信用電子メールアドレスとする。

【００９１】

その後、アップロードキュー情報テーブルの該当レコード中、センターアップロードFLGがONされる（ステップ１４０５）。

【００９２】

次に、クライアントＰＣ１２から送信されたメールを受信した制御サーバ１０のメール処理部２６およびファイル管理部２４の処理について、図１５を参照して説明する。

【００９３】

まず、メール処理部２６が受信したメールの送信元電子メールアドレスを抽出し、ファイル管理部２４が、オンラインノード管理テーブルを参照して、送信元のノード名などを特定する（ステップ１５０１）。より具体的には、オンラインノード管理テーブルから以下の情報が抽出される。
ノード名
仮登録FLG
ノード識別コード
通知用電子メールアドレス
ファイル受信用電子メールアドレス

【００９４】

次いで、メール処理部２６においてメールが復号化される（ステップ１５０２）。復号化された制御ファイルから、ファイル名等から特定されるレコードが、実ファイル情報テーブル中に存在するか否かを判断する（ステップ１５０３）。存在しない場合には（ステップ１５０３でノー(No)）、実ファイル情報テーブルおよび実ファイル世代情報テーブルの新たなレコードを作成するとともに（ステップ１５０４）、ファイルをＤＢ２０中の所

定の領域に記憶する（ステップ１５０５）。

【００９５】

レコードが実ファイル情報テーブル中に存在する場合には（ステップ１５０３でイエス(Yes)）、ファイル管理部２４は、実ファイル世代情報の該当レコードを参照して、同一ハッシュ値のレコードが存在するか否かを判断する（ステップ１５０６）。存在しない場合（ステップ１５０６でノー(No)）には、実ファイル世代情報に新たなレコードを追加するとともに（ステップ１５０７）、ファイルをＤＢ２０中の所定の領域に記憶する（ステップ１５０８）。

【００９６】

このような処理の後、アップロードメールを送信してきたクライアントＰＣ１２に、アップロード完了通知メールが送信される（ステップ１５０９）。

【００９７】

さらに、ファイル管理部２４は、利用者ディレクトリ情報テーブルを参照して、メール送信されたファイルのノード名、ディレクトリ名と同一のノード名（所有ノード名）、ディレクトリ名（利用ディレクトリ名）を含むレコードが存在するか否かを判断する（ステップ１５１０）。存在する場合（ステップ１５１０でイエス(Yes)）、当該レコード中の利用ノード名に基づき、利用者を特定して、メール処理部２６が、そのクライアントＰＣ宛（ファイル受信用電子メールアドレス宛）に、ダウンロードメールを作成して送信する。（ステップ１５１１）。

【００９８】

アップロード完了通知のメールを受信したクライアントＰＣ１２においては、アップロードキュー情報テーブル中の該当レコードが削除され、また、当該削除されるレコード中のファイル別名を持つファイルが、アップロード／ダウンロード作業ディレクトリから削除される。

【００９９】

なお、上述した説明においては、所有者のクライアントＰＣ（たとえば、クライアントＰＣ１２－１）においてファイルが作成、変更され、そのファイルがアップロードされる例を示した。これに加えて、所有者のディレクトリが公開可でかつ更新可であれば、利用者のクライアントＰＣ（たとえば、クライアントＰＣ１２－２）が、ファイルを作成、更新することも生じる。この場合にも、ほぼ同様の手順でファイルがアップロードされる。この場合、ダウンロードメールは、ファイルの所有者および他の利用者のクライアントＰＣに対して送信される。

【０１００】

次に、クライアントＰＣによる、制御サーバ１０からのダウンロード処理について説明する。制御サーバ１０は、ファイルが作成、変更されたことを通知するとともに、作成、変更されたファイルを、クライアントＰＣに送信するために、以下の情報を含むメールを作成する。

情報数：１（１つのメールに幾つの情報が入っているかを示す。この場合は１つ）

メールタイプ：アップロード完了メールである事を示すキーワード（例えばUPLOADDONE）

アップロード検知日時：アップロードメールの該当項目

所有ノード名：アップロードメールの該当項目

所有ディレクトリ呼称：アップロードメールの該当項目

ファイル名：アップロードメールの該当項目

ファイルハッシュ値：アップロードメールの該当項目

なお、上記情報およびファイルは、受信者の鍵で暗号化されて、メールに添付される。

【０１０１】

クライアントＰＣ１２のメール処理部３６は、受信したメールの添付ファイルを復号化する。ファイル管理部３４は、所有ディレクトリ情報テーブル、利用ディレクトリテーブルを参照して、メール中の所有ノード名、所有ディレクトリ名を含むレコードが存在する

か否かを判断する。該当レコードが存在すれば、記憶装置３０中の該当するディレクトリに添付されたファイルを保存する。

【０１０２】

本実施の形態においては、データ交換に電子メールを使用する。したがって、未配達という事態が発生し得る。また、通常の電子メールアドレスを使用してダウンロードメールを受取るユーザは、受信したメールの添付ファイルをクリックして、クライアントＰＣの記憶装置（ローカルディスク）にファイル等を登録する手順を忘れてしまう場合がある。

【０１０３】

このため、一定間隔（たとえば、１回／日、１回／週、１回／月）で、記憶装置（ローカルディスク）の状態とコントロールセンタの状態を比較し、異なる部分に補完する作業が必要となる。

【０１０４】

処理タイミングが到来すると、クライアントＰＣ１２の同期処理部３２は、所有ディレクトリ情報テーブルに登録されている所有ディレクトリ毎、および、利用ディレクトリ情報テーブルに登録されている利用ディレクトリ毎に、以下に述べる処理を実行する。なお、同期処理部３２は、アップロードキューテーブルに、未処理のレコードが存在していた場合は、未処理レコードが全てなくなるまで処理を延期する。

【０１０５】

同期処理部３２は、所有ディレクトリ情報テーブルに登録されている各レコードについて以下の処理を行う。

【０１０６】

図１６に示すように、まず、基本情報ノードテーブル中の自ノード名と、所有ディレクトリ情報テーブルのレコード中の所有ディレクトリ呼称を使用して、ローカルファイル情報テーブルを検索する（ステップ１６０１）。該当するレコードがない場合は（ステップ１６０２でノー(No)）、次の所有ディレクトリ情報テーブルのレコード処理に進む。該当レコードが存在する場合には（ステップ１６０２でイエス(Yes)）、以下の情報を含む同期チェックメール用のテキストファイルを作成する（ステップ１６０３）。

【０１０７】

情報数：１つのメールに幾つの情報が入っているかを示す（該当するレコード数）。
以下の項目（メールタイプ～ファイルサイズ）は、該当するレコード毎に作成される。
メールタイプ：同期チェックメールである事をキーワード（例えばSYNCCHECK）
ノード識別コード：基本情報ファイルテーブルの自ノード識別コード
ディレクトリタイプ：所有ディレクトリ
所有ノード名：所有ディレクトリ情報テーブルの所有者ノード名
所有ディレクトリ呼称：所有ディレクトリ情報テーブルの所有ディレクトリ呼称
相対Path：ローカルファイル情報テーブルのローカル相対Path
ファイル名：ローカルファイル情報テーブルのファイル名
ファイルハッシュ値：ローカルファイル情報テーブルのファイルハッシュ値
タイムスタンプ：ローカルファイル情報テーブルのタイムスタンプ
ファイルサイズ：ローカルファイル情報テーブルのファイルサイズ

【０１０８】

次いで、上記テキストファイルが暗号化され（ステップ１６０４）、暗号化されたファイルを添付したメールが、制御サーバ宛に送信される（ステップ１６０５）。

【０１０９】

同期処理部３２は、利用ディレクトリ情報テーブルに登録されている各レコードについても同様の処理を実行する。簡単に説明すると、利用ディレクトリ情報テーブルの所有者ノード名と所有ディレクトリ呼称を使用して、ローカルファイル情報テーブルを検索する（ステップ１６１１）。該当するレコードがない場合は、次の利用ディレクトリ情報テーブルのレコード処理に進む。

【０１１０】

該当レコードが存在する場合には、以下の情報を含む同期チェックメール用のテキストファイルを作成する。

【０１１１】

情報数：１つのメールに幾つの情報が入っているかを示す（該当するレコード数）。
以下の項目（メールタイプ～ファイルサイズ）は、該当するレコード毎に作成される。
メールタイプ：同期チェックメールである事をキーワード（例えばSYNCCHECK）
ノード識別コード：基本情報ファイルテーブルの自ノード識別コード
ディレクトリタイプ：利用ディレクトリ
所有ノード名：利用ディレクトリ情報テーブルの所有者ノード名
所有ディレクトリ呼称：利用ディレクトリ情報テーブルの所有ディレクトリ呼称
相対Path：ローカルファイル情報テーブルのローカル相対Path
ファイル名：ローカルファイル情報テーブルのファイル名
ファイルハッシュ値：ローカルファイル情報テーブルのファイルハッシュ値
タイムスタンプ：ローカルファイル情報テーブルのタイムスタンプ
ファイルサイズ：ローカルファイル情報テーブルのファイルサイズ

【０１１２】

次いで、上記テキストファイルが暗号化され、暗号化されたファイルを添付したメールが、制御サーバ宛に送信される。

【０１１３】

同期チェックメールを受信した制御サーバ１０における処理について以下に説明する。

【０１１４】

図１７に示すように、制御サーバ１０のメール処理部２６がメールを受信すると、同期処理部２２が、メールから抽出された送信元電子メールアドレスが、オンラインノード管理テーブルに登録されているかチェックし、オンラインノード管理テーブルの該当レコードから、以下の情報を取得する（ステップ１７０１）。

ノード名
仮登録FLG
ノード識別コード
通知用電子メールアドレス
ファイル受信用電子メールアドレス

【０１１５】

また、メール処理部２６は、添付されたテキストファイルを復号化する（ステップ１７０２）。同期処理部２２は、復号化されたメールから、以下の項目を抽出する（ステップ１７０３）。

所有ノード名
所有ディレクトリ呼称
相対Path
ファイル名

【０１１６】

同期処理部２２は、これらを含むレコードが、実ファイル情報テーブルに存在するか否かを判断する（ステップ１７０４）。実ファイル情報テーブルにレコードが存在する場合には（ステップ１７０４でイエス(Yes)）、実ファイル世代情報テーブルの該当レコードの最新のハッシュ値と、メール中のファイルハッシュ値とを比較する（ステップ１７０５）。双方のハッシュ値が同一であれば（ステップ１７０６でイエス(Yes)）、同期チェックメールを送信したクライアントＰＣにあるファイルと、制御サーバ１０にあるファイルとは同一であると判断する。

【０１１７】

ハッシュ値が異なる場合（ステップ１７０６でノー(No)）、同期処理部２２は、実ファイル情報テーブルの該当レコードのタイムスタンプと、メール中のタイムスタンプとを比較する（ステップ１７０７）。メール中のタイムスタンプが新しい場合（ステップ１７０

８でイエス(Yes)）には、同期チェックメールを送信したクライアントＰＣに、アップロード依頼メールを送信する（ステップ１７０９）。その一方、メール中のタイムスタンプが古い場合（ステップ１７０８でノー(No)）、同期チェックメールを送信したクライアントＰＣに、ダウンロードメールを送信する（ステップ１７１０）。

【０１１８】

レコードが実ファイル情報テーブルに存在しない場合には（ステップ１７０４でノー(No)）、図１８に示すように、同期処理部２２は、メール中のディレクトリタイプが所有ディレクトリである場合（ステップ１８０１でイエス(Yes)、または、メール中のディレクトリタイプが利用ディレクトリで（ステップ１８０２でイエス(Yes)）、かつ、所有ディレクトリ情報テーブル中、関連するレコードの「公開フラグ＝公開」かつ「公開タイプ＝更新可」であれば（ステップ１８０３でイエス(Yes)）、同期チェックメールを送信したクライアントＰＣに、アップロード依頼メールを送信する（ステップ１８０４）。

【０１１９】

さらに、メール本文中、全てのファイルについて、上記図１７および図１８に示す処理が実行された後、実ファイル情報テーブル中に、同期処理が行われていないレコードが残った場合には、当該レコードに関して、所有者のクライアントＰＣ宛にダウンロードメールを送信する。

【０１２０】

制御サーバ１０におけるダウンロードメール、および、アップロード依頼メールの作成について以下に簡単に説明する。ダウンロードメールは、以下の情報を含むテキストファイルを添付した状態で、クライアントＰＣに送信される。

情報数：１（１つのメールに幾つの情報が入っているかを示す。この場合には１つ）

メールタイプ：ダウンロードメールである事を示すキーワード（例えばDOWNLOAD）

所有ノード名：実ファイル情報テーブルの所有ノード名

所有ディレクトリ呼称：実ファイル情報テーブルの所有ディレクトリ呼称

ファイル名：実ファイル情報テーブルの実ファイル名

ファイルハッシュ値：実ファイル世代情報テーブルのハッシュ値

添付ファイル名：実ファイル世代情報テーブルのサーバーファイル名

【０１２１】

また、アップロード依頼メールは、以下の情報を含むテキストファイルを添付した状態で、クライアントＰＣに送信される。

情報数：１（１つのメールに幾つの情報が入っているかを示す。この場合１つ）

メールタイプ：アップロード依頼メールである事を示すキーワード（例えばREQUPLOAD）

所有ノード名：同期チェックメール本文内の所有者ノード名

所有ディレクトリ呼称：同期チェックメール本文内の所有ディレクトリ呼称

相対Path：同期チェックメール本文内のローカル相対Path

ファイル名：同期チェックメール本文内のファイル名

クライアントＰＣにおけるファイルのダウンロード処理については、前述したとおりである。また、アップロード依頼に応じたアップロード処理も、基本的には、アップロード処理と同様である。

【０１２２】

次に、本実施の形態の適用事例について説明する。まず、制御サーバ１０を、クライアントＰＣ１２のファイルバックアップ装置として使用する場合について説明する。図１０の例で、クライアントＰＣ１２－１において、あるファイル（クライアントＰＣ１２－１がファイル所有者となっているファイル）が更新されたと考える。クライアントＰＣ１２－１は、更新されたファイルを含むメールを制御サーバ１２に送信する（ステップ１１０２）。制御サーバ１０は、ファイルの変更を検知した後（図１２、図１３参照）、ファイルが添付されたメールを、制御サーバ１０に送信する。制御サーバ１０は、送信されたメールに添付されたファイルを、ＤＢ２０中の所定の位置に格納するとともに、実ファイル

世代情報テーブルのレコードを更新する。これにより、データのバックアップを実現することができる。

【０１２３】

次に、チームで作業をする場合について説明する。あるクライアントＰＣ（たとえば、クライアントＰＣ１２－１）がライブラリアンとして機能し、他のクライアントＰＣ（たとえば、クライアントＰＣ１２－２、１２－３、・・・）が、チームのメンバーとして機能すると考える。

【０１２４】

システム設計など、複数人で１つのプロジェクトに関する仕事をする場合などでは、いつだれがどのファイルを作成・変更したかの記録と作成・変更タイミングの衆知徹底がプロジェクト推進上重要なポイントとなる。このような場合は本機構の補助手段としてE-Mailのメーリングリスト機能をもったメールサーバ１６を組み合わせることにより、ファイルを共有化した作業を実現できる。

【０１２５】

まず、ドキュメント全体を管理する立場の人（ライブラリアン）が所有ディレクトリを定義して、ライブラリアンのクライアントＰＣ１２－１が、所有ディレクトリ情報テーブルのレコードを、制御サーバ１０に登録する。これにより、制御サーバ１０には、必要な所有者Path情報テーブルのレコードが生成される。

【０１２６】

実際の作業に携わるプロジェクトのメンバーは他のメンバーが作成・修正したファイルを制御サーバ１０から受信して、自動的に、自己のクライアントＰＣのローカルディスクに保管することができ、かつ、自己のクライアントＰＣにおいて、ファイルの変更が検知されると、当該クライアントＰＣから変更されたファイルが添付されたメールを、制御サーバ１０に送信する。

【０１２７】

また、プロジェクトが一区切りついたとき（例えば、製品の１次リリース向けの作業完了時）には、ライブラリアンがその所有ディレクトリを公開・更新可から公開・更新不可と属性変更することにより、ドキュメントの凍結を行う事が簡単にできる。

【０１２８】

また、そのリリース時点のファイルを固定・保管する為に、ライブラリアンが利用ディレクトリを定義し、一旦固定したファイルをローカルディスクに、ダウンロードし、その後、改めてそのローカルディスクのディレクトリを先に登録した、所有ディレクトリとは別の所有ディレクトリ（公開・更新不可）として定義する事により、特定時点のファイルを永久に保管する事が可能である。その後、元の所有ディレクトリを公開・更新可と属性変更することにより、次のリリースに向けた変更作業が継続できるようになる。

【０１２９】

たとえば、図１１に示すように、ライブラリアンのクライアントＰＣ１２－１は、共有するファイルの所有ディレクトリ情報テーブルのレコードを、制御サーバ１０に送信する。ここで、レコード中、「公開フラグ」は「ON」、「公開タイプ」は「更新可」としておく。送信されたレコードは、制御サーバ１０において、所有者パス情報テーブルのレコードとして登録される。

【０１３０】

その一方、プロジェクトのメンバーのクライアントＰＣ１２－２、１２－３、・・・においては、制御サーバ１０の所有者パス情報テーブルのレコードを参照して、クライアントＰＣの記憶装置中に、利用ディレクトリ情報テーブルのレコードを作成しておく。

【０１３１】

たとえば、プロジェクトのメンバーのクライアントＰＣ１２－２は、利用ディレクトリ情報テーブルのレコードにて特定されるディレクトリのファイルを作成、更新する。ファイルが作成、更新されると、クライアントＰＣ１２－２は、更新されたファイルを添付したメールを、制御サーバ１０に送信する。

【０１３２】
制御サーバ１０は、ファイルをＤＢ２０に記憶するとともに、実ファイル世代情報の該当レコードを更新する。また、制御サーバ１０は、ファイルを添付したメールを、他のクライアントＰＣ１２－１、１２－３、・・・に送信する。
【０１３３】
次に、ファイルの凍結作業と次のフェーズの作業開始について説明する。クライアントＰＣ１２－１は、所有ディレクトリ情報テーブルの該当ファイルのレコードにおいて、「公開タイプ」を「更新可」から「更新不可」に変更する。なお、これに先立って、ライブラリアンのクライアントＰＣ１２－１は、変更作業停止の通知を、メールなどの手段により、クライアント１２－２、１２－３、・・・に通知しても良い。
【０１３４】
その後、クライアントＰＣ１２－１は、凍結ファイル格納用のディレクトリを、記憶装置３０中に新規作成し、そのディレクトリを、利用ディレクトリとして登録する。つまり、利用ディレクトリ情報テーブルのレコードを作成する。これにより、制御サーバ１０から、各ファイルの最新バージョンのものが全て、クライアントＰＣ１２－１に、添付ファイルとしてメールにて送信される。
【０１３５】
メールの受信後、つまり、ダウンロード終了後、クライアントＰＣ１２－１において、利用者ディレクトリ情報テーブルのレコードを削除し、改めて、そのローカルディレクトリについて、所有ディレクトリとして定義する。つまり、そのローカルディレクトリの情報を、所有ディレクトリ情報テーグルのレコードとして格納する。ここでは、「公開タイプ」を「更新不可」としておく。これにより、このローカルディレクトリのファイルは、別途、制御サーバ１０で管理されることになる。
【０１３６】
その後、当初のディレクトリの「公開タイプ」を、「更新不可」から「更新可」に変更することで、利用者によるファイルの更新が可能となる。たとえば、クライアントＰＣ１２－１は、変更作業再開の通知を、メールにて、クライアント１２－２、１２－３に通知しても良い。
【０１３７】
上述したクライアントＰＣ１２－１の処理の結果、制御サーバ１０のＤＢ２０には、各メンバーが継続して作業するための所有ディレクトリ（もともと作業に使用していたディレクトリ）と、リリースに伴って固定されたファイルが格納されている変更不可の所有ディレクトリの２つが存在することになる。
【０１３８】
本発明は、以上の実施の形態に限定されることなく、特許請求の範囲に記載された発明の範囲内で、種々の変更が可能であり、それらも本発明の範囲内に包含されるものであることは言うまでもない。
【図面の簡単な説明】
【０１３９】
【図１】図１は、本発明の実施の形態にかかるファイルバックアップ・共有制御サーバを含むシステム全体の構成を示すブロックダイヤグラムである。
【図２】図２（ａ）は制御サーバの構成を示すブロックダイヤグラム、図２（ｂ）は、クライアントＰＣの構成を示すブロックダイヤグラムである。
【図３】図３は、本実施の形態にかかるノード基本情報テーブルのデータ構成を示す図である。
【図４】図４は、本実施の形態にかかる所有ディレクトリ情報テーブルのデータ構成を示す図である。
【図５】図５は、本実施の形態にかかる利用ディレクトリ情報テーブルのデータ構成を示す図である。
【図６】図６（ａ）は、本実施の形態にかかるローカルファイル情報テーブルのデータ構

成を示す図、図６（ｂ）は、アップロードキュー情報テーブルのデータ構成を示す図である。
【図７】図７は、本実施の形態にかかるオンラインノード管理テーブルのデータ構成を示す図である。
【図８】図８は、本実施の形態にかかる所有者情報テーブルのデータ構成を示す図である。
【図９】図９（ａ）は、本実施の形態にかかる利用者情報テーブルのデータ構成を示す図、図９（ｂ）は、実ファイル情報テーブルのデータ構成を示す図、図９（ｃ）は、実ファイル世代情報テーブルのデータ構成を示す図である。
【図１０】図１０は、本実施の形態において、制御サーバをファイルバックアップ装置として利用する例を示す図である。
【図１１】図１１は、本実施の形態において、制御サーバをファイル共有装置として利用する例を示す図である。
【図１２】図１２は、本実施の形態にかかるクライアントＰＣにて実行される処理を示すフローチャートである。
【図１３】図１３は、本実施の形態にかかるクライアントＰＣにて実行される処理を示すフローチャートである。
【図１４】図１４は、本実施の形態にかかるクライアントＰＣにて実行される処理を示すフローチャートである。
【図１５】図１５は、本実施の形態にかかる制御サーバにて実行される処理を示すフローチャートである。
【図１６】図１６は、本実施の形態にかかるクライアントＰＣにて実行される処理を示すフローチャートである。
【図１７】図１７は、本実施の形態にかかる制御サーバにて実行される処理を示すフローチャートである。
【図１８】図１８は、本実施の形態にかかる制御サーバにて実行される処理を示すフローチャートである。
【符号の説明】
【０１４０】
１０　制御サーバ
１２　クライアントＰＣ
１４　ネットワーク
１６　メールサーバ
２０　ＤＢ
２２　同期処理部（センター同期処理部）
２４　ファイル管理部（センターファイル管理部）
２６　メール処理部（センターメール処理部）
３０　記憶装置
３２　同期処理部（ローカル同期処理部）
３４　ファイル管理部（ローカルファイル管理部）
３６　メール処理部（ローカルメール処理部）

【図１】

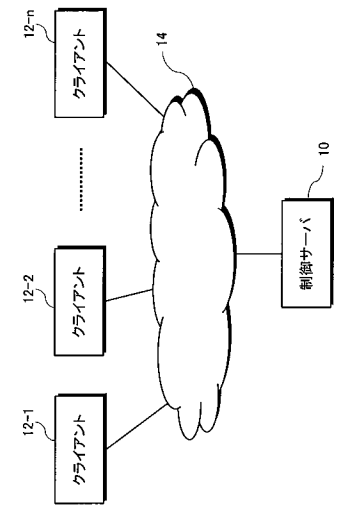


図1

【図２】

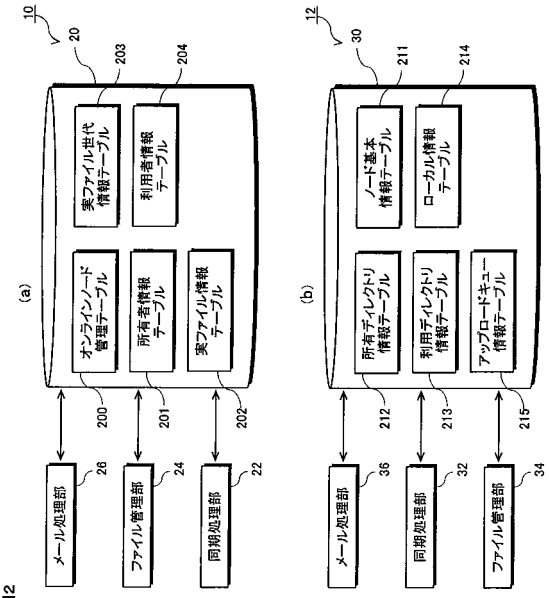


図2

【図３】

図3

項目名	内容説明
自ノード名	サポートプログラムを使用して手動登録する
自ノード識別コード	センターコントローラが決めた認証コード（ノード識別パスワード）
PGP暗号Passフェーズ	自身のPGP暗号鍵登録時に登録したPassフェーズ
センター電子メールアドレス	センターコントローラの基本制御電子メールアドレス
ユーザータイプ	
SMTPサーバー名	サポートPGMが自動メール発信する場合に使用できるSMTPサーバーのHOST名（もしくはIPアドレス）
自動受信用メールサーバー名	センターコントローラからのファイル送信メールをサポートPGMが自動で受信してもよい環境の場合、そのメールサーバーのHOST名（もしくは、IPアドレス）
通知受信用電子メールアドレス	センターコントローラから各種通知メールを受信する時に使用する電子メールアドレス。複数ある場合は、各アドレスをカンマで区切って登録（携帯電話のメールアドレスも可）。ノードPC側から自動発信するメールのFROMにも使用する。
ファイル受信用電子メールアドレス	実ファイル（暗号化後）を添付ファイルとして受信する場合に使用するメールアドレス（携帯電話のメールアドレスは不可）
ファイル受信用アカウント	ファイル受信用電子メールを受信する時にメールサーバーへの接続に使用するアカウント
ファイル受信用パスワード	ファイル受信用電子メールを受信する時にメールサーバーへの接続に使用するパスワード
メール自動発信時アドレス	サポートPGMがメールを自動で発信する場合にFROM:欄に使用するメールアドレス
センターコントローラID	暗号鍵ファイルに登録したセンター公開鍵のKeyとして使用する
アップロード/ダウンロード作業ディレクトリ	アップロード/ダウンロード作業時に使用する作業ディレクトリPath
前回SyncCheck日時	前回コントロールセンタとの同期処理を行った日時

【図４】

図4

項目名	内容説明
所有ディレクトリPath	センターコントローラ側にバックアップを取得するローカルディスクのディレクトリPath。登録したディレクトリ配下の全てのファイル、サブディレクトリーがバックアップ対象となる。
削除FLG	ON:ディレクトリ所有者によって削除された OFF:削除されていない
所有ディレクトリ呼称	所有ディレクトリの別名（ユニークキー） （ネットワークに公開する場合に公開Keyの一部として使用される）
センター通知FLG	ON:センターコントローラに所有ディレクトリの通知を実施済み OFF:センターコントローラに所有ディレクトリの通知を未実施
センター受領FLG	ON：センターコントローラから所有ディレクトリ情報登録確認が来た OFF:センターコントローラから所有ディレクトリ情報登録確認が来ていない
最大保存世代数	センターコントローラ側に取得するバックアップファイルの世代の最大数 新規世代と判定する基準は、ファイル内容が変更されたタイミング
削除ファイル保存期間	ディレクトリ所有者によってローカルディスクからファイルが削除された場合でもセンターコントローラ側でそのファイルを保管している期間。 未登録の場合、ディレクトリ所有者が削除したタイミングでセンターコントローラ側のファイルも削除される。
公開フラグ	ON:　ネットワークに公開する。 OFF:　ネットワークに公開しない。
公開パスワード	公開フラグ＝ONの時必須、=OFFのとき登録不要 ディレクトリ利用者がネットワーク内に公開されたディレクトリを利用する場合に認証する為のパスワード
公開タイプ	公開フラグ＝ONの時必須、=OFFのとき登録不要 更新可:公開したファイルをディレクトリ利用者が書き換えた場合にディレクトリ所有者に結果を反映する 更新不可:公開したファイルをディレクトリ利用者が書き換えた場合にディレクトリ所有者に結果を反映しない
通知メール受領フラグ	公開タイプが「更新可」の場合に登録可能 所有ディレクトリに変更が発生した場合にセンターコントローラから通知電子メールを受領する場合はON、受領しない場合はOFFとする。
通知メール受信タイプ	通知メール受領フラグがONの場合にのみ登録する。 １：センターコントローラ側でファイルの更新が発生した都度送信 ２：1日に1回更新情報をまとめて通知

【図５】

図5

項目名	内容説明
所有者ノード名	利用ディレクトリの所有者ノード名（ユニークキー）
所有ディレクトリ呼称	所有ディレクトリの別名（ユニークキー）
公開パスワード	利用ディレクトリの公開パスワード
更新可否FLG	ON：利用ディレクトリが更新可で公開されている場合 OFF：利用ディレクトリが更新不可で公開されている場合
ファイル格納ローカルPath	利用ディレクトリのファイルを格納するローカルディスクPath
センター通知FLG	ON:センターコントローラに利用ディレクトリの通知を実施済み OFF:センターコントローラに利用ディレクトリの通知を未実施
センター受領FLG	ON:センターコントローラから利用ディレクトリ情報登録確認が来た OFF:コントロールセンタから利用ディレクトリ情報登録確認が来ていない

【図６】

図6

（a）

項目名	内容説明
所有者ノード名	ユニークキー
所有ディレクトリ呼称	ユニークキー
ローカル相対Path	ユニークキー 所有ディレクトリ、利用ディレクトリとして指定したローカルディスク内のPathを基準とした相対Path
ファイル名	ユニークキー
ファイルハッシュ値	ローカルファイルのMD5ハッシュ値
タイムスタンプ	ファイルの更新時刻
ファイルサイズ	Byte数
アップロード検知作業中FLG	ON：検知作業中 OFF：検知作業未作業

（b）

項目名	内容説明
アップロード検知日時	ユニークキー
所有ノード名	ユニークキー
所有ディレクトリ呼称	ユニークキー
ファイル名	ユニークキー
ファイルハッシュ値	ユニークキー
タイムスタンプ	ファイルの更新時刻
ファイルサイズ	Byte数
削除FLG	ON:ローカルディスクから削除された OFF:削除されていない
センターアップロードFLG	ON:センターコントローラに該当ファイルをアップロード済み OFF:センターコントローラに該当ファイルを未アップロード
ファイル別名	アップロード対象ファイルと検知された時点で、アップロード作業ディレクトリに別名でコピーされる。そのコピー後の別名。但し、削除されたファイルと検知された場合は、未登録

【図７】

図7

項目名	内容説明
ノード名	ユニークキー
仮登録FLG	ON/OFF ON:メンテナンスPGMにて手動登録しただけで、ノードPCとの通信が確立していない。 OFF:ノードPCとの通信が確立した（ノードPCからHELODONEメールが届いた） 仮登録FLGがONの間は、該当ノードへの配信は行われない。また、該当ノードからのファイルアップロードやディレクトリ登録などの依頼も受付ない
ノード識別コード	ノード名登録時に自動生成された識別用パスワード
ユーザータイプ	
ノード発信電子メールアドレス	ノード側のメール自動発信時電子メールアドレス センターコントローラ宛に届いたメールの正当性チェックの第一段階で使用
通知用電子メールアドレス	
ファイル受信用電子メールアドレス	
ノード登録日	
受信頻度タイプ	
ノード登録申請者名	

【図８】

図8

項目名	内容説明
ノード名	必須（ユニークキー）
所有ディレクトリ呼称	所有ディレクトリの別名（ユニークキー）
削除FLG	ON:ディレクトリ所有者によって削除された OFF:削除されていない
削除日時	削除FLG＝ONの場合記録 ディレクトリ所有者から削除（DelDir）の通知が来た日時を記録 （物理的に削除する時、保存期間とのチェックに使用する）
最大保存世代数	センターコントローラ側に取得するバックアップファイルの世代の最大数 新規世代と判定する基準は、ファイル内容が変更されたタイミング
削除ファイル保存期間	ディレクトリ所有者によってローカルディスクからファイルが削除された場合でもセンターコントローラ側でそのファイルを保管している期間。 未登録の場合、ディレクトリ所有者が削除したタイミングでセンターコントローラ側のファイルも削除される。
公開フラグ	ON:　ネットワークに公開する。 OFF:　ネットワークに公開しない。
公開パスワード	公開フラグ＝ONの時必須、=OFFのとき登録不要 ディレクトリ利用者がネットワーク内に公開されたディレクトリを利用する場合に認証する為のパスワード
公開タイプ	公開フラグ＝ONの時必須、=OFFのとき登録不要 更新可：　公開したファイルをディレクトリ利用者が書き換えた場合にディレクトリ所有者に結果を反映する 更新不可：公開したファイルをディレクトリ利用者が書き換えた場合にディレクトリ所有者に結果を反映しない
通知メール受領フラグ	公開タイプが「更新可」の場合に登録可能 所有ディレクトリに変更が発生した場合にセンターコントローラから通知電子メールを受領する場合はON、受領しない場合はOFFとする。
通知メール受信タイプ	通知メール受領フラグがONの場合にのみ登録する。 １：センターコントローラ側でファイルの更新が発生した都度送信 ２：1日に１回更新情報をまとめて通知

【図９】

図9

(a)

項目名	内容説明
利用ノード名	ユニークキー
所有ノード名	利用している所有ディレクトリの所有者のノード名（ユニークキー）
利用ディレクトリ呼称	利用している所有ディレクトリの別名（ユニークキー）
削除FLG	ON:ディレクトリ利用者によって削除された OFF:削除されていない

(b)

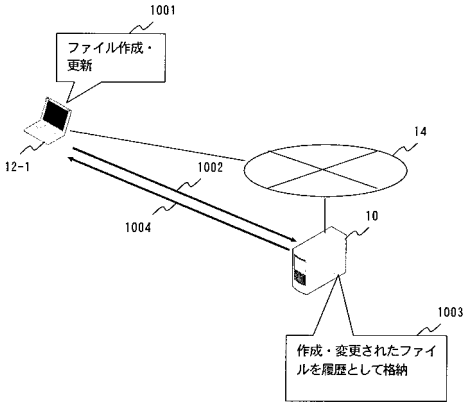
項目名	内容説明
所有ノード名	各ノードから通知されたノード名（ユニークキー）
所有ディレクトリ呼称	各ノードから通知された所有ディレクトリ呼称（ユニークキー）
相対Path	ノード側で所有ディレクトリ内のサブディレクトリにファイルが格納されている場合、所有ディレクトリからの相対Path（ユニークキー）
実ファイル名	ノード側で格納されていたファイル名（ユニークキー）
サーバーファイル名	センターコントローラ側に格納する場合のファイル名（除く拡張子） センターコントローラ全体で登録の都度ユニークな名称を採番して登録する
削除フラグ	ON: ノード側で削除された OFF：削除されていない

(c)

項目名	内容説明
サーバーファイル名	ユニークキー
登録日時	センターコントローラに保管した日時(センターコントローラのマシン時刻) センターコントローラ内の物理ファイル名は、実ファイル名.登録日時として格納される（ユニークキー）
登録ディレクトリ	センターコントローラにファイルが格納されているPath 格納Dir情報に登録されているセンターコントローラ内Pathの１つ
登録ノード名	格納したファイルを入手したノード名
格納ファイルサイズ	センターコントローラ内に格納された状態（圧縮後）のファイルサイズ
ファイルサイズ	圧縮解凍後（ノードPC側）のファイルサイズ
ハッシュ値	ノード側で格納されていた状態のファイルのハッシュ値
圧縮後ハッシュ値	ノード側で圧縮後のファイルのハッシュ値
タイムスタンプ	ノード側で格納されていたファイルのファイルタイムスタンプ

【図１０】

図１０



【図１１】

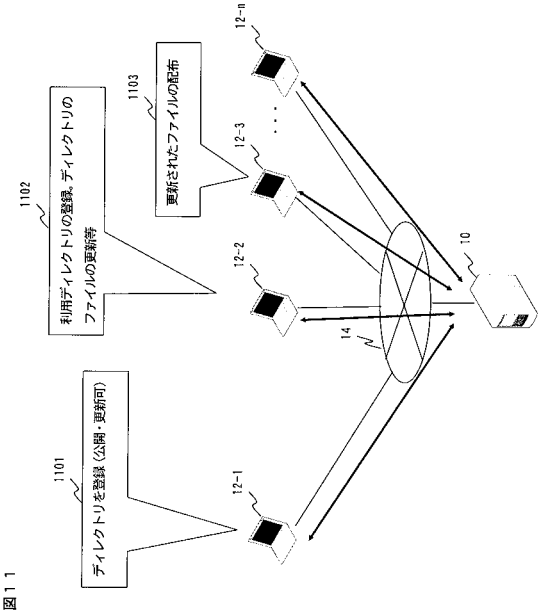
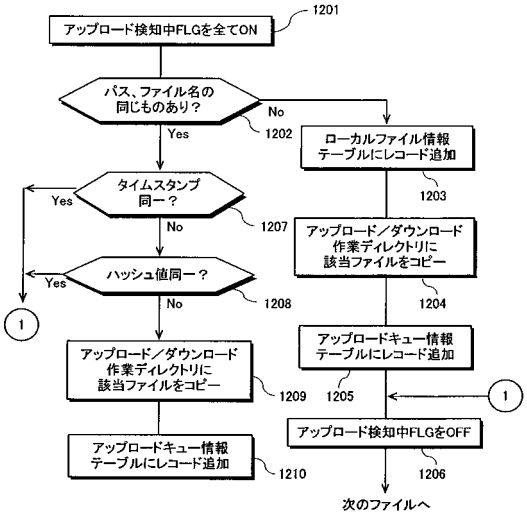


図１１

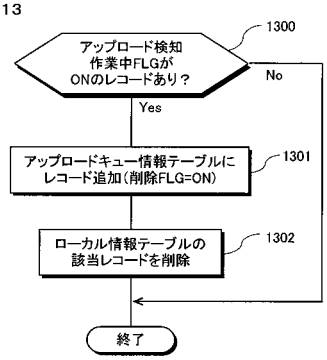
【図１２】

図12



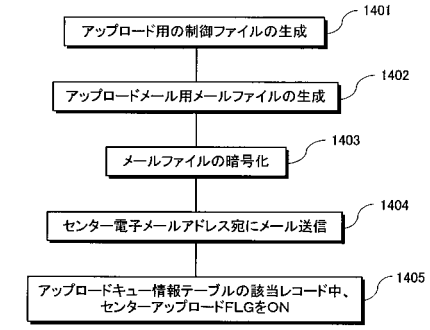
【図１３】

図13



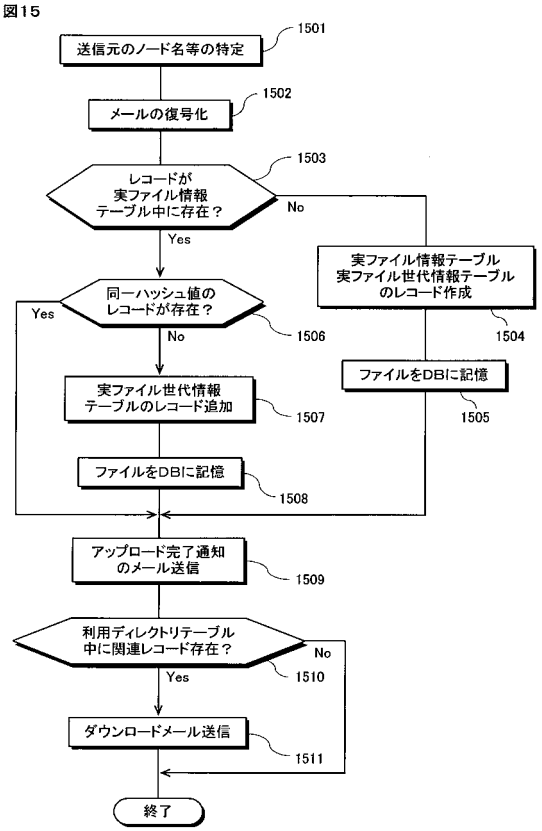
【図１４】

図14



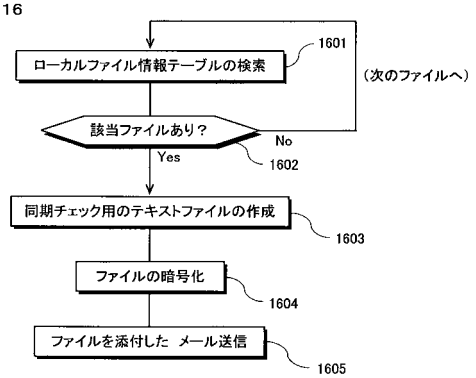
【図１５】

図15



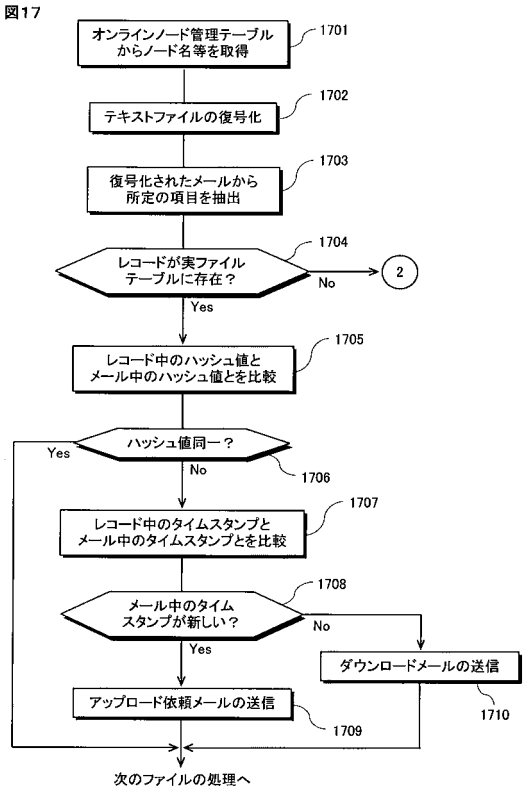
【図１６】

図16



【図１７】

図17



【図１８】

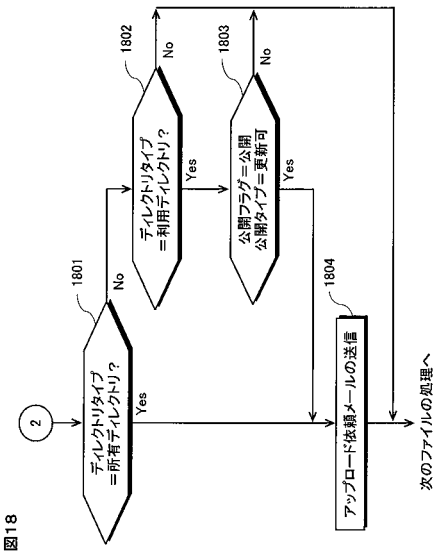

2
ディレクトリタイプ
＝所有ディレクトリ？
1801
No
Yes
ディレクトリタイプ
＝利用ディレクトリ？
1802
No
Yes
公開フラグ＝公開
公開タイプ＝更新可
1803
No
Yes
アップロード依頼メールの送信
1804
次のファイルの処理へ

図18

フロントページの続き

(51)Int.Cl.[7]	FI	テーマコード（参考）
	G06F 13/00 630A	