(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6668948号
(P6668948)
(45)発行日　令和2年3月18日(2020.3.18)　(24)登録日　令和2年3月2日(2020.3.2)

(51)Int.Cl.　　F I
*G06F 11/34 (2006.01)*　　G06F 11/34 176
*G06F 11/30 (2006.01)*　　G06F 11/30 140M

請求項の数 9　(全 19 頁)

(21)出願番号　特願2016-106657(P2016-106657)
(22)出願日　平成28年5月27日(2016.5.27)
(65)公開番号　特開2017-211945(P2017-211945A)
(43)公開日　平成29年11月30日(2017.11.30)
審査請求日　平成31年2月12日(2019.2.12)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100107766
弁理士　伊東　忠重
(74)代理人　100070150
弁理士　伊東　忠彦
(74)代理人　100192636
弁理士　加藤　隆夫
(72)発明者　千葉　嗣部
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

最終頁に続く

(54)【発明の名称】ファイル判定プログラム、ファイル判定装置およびファイル判定方法

(57)【特許請求の範囲】
【請求項１】
属性情報に特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与し、
付与された前記順位に従って、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かによりファイルの種別に関する判定を行う、
処理をコンピュータに実行させることを特徴とするファイル判定プログラム。
【請求項２】
前記属性情報は、ファイルの所在を示す情報である、
ことを特徴とする請求項１に記載のファイル判定プログラム。
【請求項３】
前記種別に関する判定結果をファイルの属性情報に対応付けて記憶する、
ことを特徴とする請求項１または２に記載のファイル判定プログラム。
【請求項４】
前記種別に関する判定結果を、さらに、実行中のプロセスに対応付けて記憶する、
ことを特徴とする請求項３に記載のファイル判定プログラム。
【請求項５】
複数の実行中のプロセスそれぞれについて複数の監視プロセスが処理を行う第１のステップと、
一巡目の処理を終了した監視プロセスの処理を他の監視プロセスが引き継いで以後の処

理を行う第２のステップと、
をコンピュータに実行させることを特徴とする請求項１乃至４のいずれか一項に記載のファイル判定プログラム。
【請求項６】
前記第２のステップは、担当する前記プロセスのいずれかのファイル識別子数の増加を検知した場合に、該プロセスについての処理を開始する、
ことを特徴とする請求項５に記載のファイル判定プログラム。
【請求項７】
前記属性情報に特定の文字列が含まれるか否かの判定は、ファイル名にログであることを示す所定の文字列が含まれているか否かにより行う、
ことを特徴とする請求項１乃至６のいずれか一項に記載のファイル判定プログラム。
【請求項８】
属性情報に特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与する手段と、
付与された前記順位に従って、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かによりファイルの種別に関する判定を行う手段と、
を備えたことを特徴とするファイル判定装置。
【請求項９】
属性情報に特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与し、
付与された前記順位に従って、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かによりファイルの種別に関する判定を行う、
処理をコンピュータが実行することを特徴とするファイル判定方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、ファイル判定プログラム、ファイル判定装置およびファイル判定方法に関する。
【背景技術】
【０００２】
コンピュータで実行中のプロセスが参照または更新するファイルが特定の種別であるか否かを判定する技術が存在する。
【０００３】
例えば、アプリケーションサーバ等で実行されるアプリケーションプロセスの出力するログを収集する業務にあっては、アプリケーションプロセスが出力する多数のファイルの中からログとみなされるファイルの特定が必要となる場合がある。ログは、アプリケーションに何らかの問題が発生した場合に原因究明を行う上で重要な情報であり、近年、内部統制への関心が高まっている中、アプリケーションが出力するログを収集することの重要性が高まっている。
【０００４】
ログを収集する業務にあっては、ログの出力先（ファイル名を含むフルパス）、フォーマットおよび収集先（抽出した内容の保存先）等を収集ツール（製品：Fluentd、Splunk等）に指定する必要がある。この際、ログの出力先等に変動がない場合は、最初に一度だけ指定を行えばよいが、アプリケーションの追加や修正によってシステム構成が変わった場合にはログの出力先等にも変動が生じ、その都度に指定をやり直さなければならない。アプリケーションには、例えば、オープンソース（ＯＳＳ）、ミドルウェア、業務アプリケーション等が含まれ、それらの追加や修正の頻度は高く、システム構成は日々変わる状況となっている。
【０００５】

ログの出力先等の指定は、管理者がアプリケーションの仕様を理解して設定を行う必要があるため、システム構成が日々変わる環境では、ログの出力先等の確認を頻繁に行わなければならず、システム管理者の負担となる。そのため、アプリケーションの出力するファイルを自動的に監視し、ログとみなされるファイルを自動的に特定することが望まれる。

【０００６】

図１は従来におけるアプリケーションの出力するファイルの監視の例を示す図であり、１つの監視プロセスにより複数のアプリプロセス（アプリケーションプロセス）のファイルを監視する例を示している。この場合、監視プロセスは、各アプリプロセスのファイル一覧からファイルを１つずつ確認していき、ファイル名や拡張子に「log」等の文字列が含まれている場合にログであると判定する。

【０００７】

図２は従来におけるアプリケーションの出力するファイルの監視の他の例を示す図であり、アプリプロセスそれぞれに対応する監視プロセスによりアプリプロセスのファイルを監視する例を示している。この場合、各監視プロセスは、対応するアプリプロセスのファイル一覧からファイルを１つずつ確認していき、ファイル名や拡張子に「log」等の文字列が含まれている場合にログであると判定する。

【０００８】

一方、アプリケーションが出力するログを収集する計算機ログ収集システムについての開示がある（特許文献１）。この場合も、アプリケーション格納場所に存在するすべてのファイルについて順次にログであるか否かの判定を行っている。

【先行技術文献】

【特許文献】

【０００９】

【特許文献１】特開２０１３－１９１０１２号公報

【発明の概要】

【発明が解決しようとする課題】

【００１０】

上述したように、従来は対象となるファイルを一元的に順次に判定の対象としていたため、ログでないものが処理順序で上位に多数存在している場合には、ログを発見するまでに時間を要するという問題があった。なお、ログを例にして説明したが、その他の種別のファイルを特定する場合も同様である。

【００１１】

そこで、一側面では、特定の種別である可能性が高いファイルを優先的に種別判定の対象ファイルとすることを目的とする。

【課題を解決するための手段】

【００１２】

一つの形態では、属性情報に特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与し、付与された前記順位に従って、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かによりファイルの種別に関する判定を行う、処理をコンピュータに実行させる。

【発明の効果】

【００１３】

特定の種別である可能性が高いファイルを優先的に種別判定の対象ファイルとすることができる。

【図面の簡単な説明】

【００１４】

【図１】従来におけるアプリケーションの出力するファイルの監視の例を示す図（その１）である。

【図２】従来におけるアプリケーションの出力するファイルの監視の例を示す図（その２）である。
【図３】情報処理装置の機能構成例を示す図である。
【図４】ログ収集システムの構成例を示す図である。
【図５】情報処理装置のハードウェア構成例を示す図である。
【図６】親監視プロセスの処理例を示すフローチャートである。
【図７】プロセス一覧およびプロセス情報の例を示す図である。
【図８】管理テーブルの例を示す図である。
【図９】子監視プロセスを起動した状態の例を示す図である。
【図１０】管理テーブルの例を示す図である。
【図１１】子監視プロセスの処理例を示すフローチャートである。
【図１２】管理テーブルの例を示す図である。
【図１３】ファイル名による簡易確認の処理例を示すフローチャートである。
【図１４】優先確認リストおよび優先確認外リストの例を示す図である。
【図１５】ファイル内容による詳細確認の処理例を示すフローチャートである。
【図１６】管理テーブルの例を示す図である。
【図１７】管理テーブルの例を示す図である。
【図１８】管理テーブルの例を示す図である。
【図１９】子監視プロセスを終了した状態の例を示す図である。
【図２０】管理テーブルの例を示す図である。
【図２１】子監視プロセスを起動した状態の例を示す図である。
【図２２】管理テーブルの例を示す図である。
【発明を実施するための形態】
【００１５】
以下、本発明の好適な実施形態につき説明する。
【００１６】
＜構成＞
図３は情報処理装置１の機能構成例を示す図である。図３において、情報処理装置１では、任意数のアプリプロセス（アプリケーションプロセス）ＡＰ１、・・が実行され、それぞれのアプリプロセスＡＰ１、・・はログや設定ファイルを参照・更新する。これらのアプリプロセスＡＰ１、・・の参照・更新するファイルがログであるか否か、すなわち、ログという特定の種別であるか否かを監視する仕組として、親監視プロセスＰＰと必要数の子監視プロセスＣＰ１、・・が設けられる。子監視プロセスＣＰ１、・・は、親監視プロセスＰＰにより起動され、代表的な子監視プロセス、例えば子監視プロセスＣＰ１を除き、所定の一巡目の処理を終了した場合に、代表的な子監視プロセスに処理を引き継いで終了する。監視の処理に際して使用されるデータとして、第１管理テーブルＴ１、第２管理テーブルＴ２、第３管理テーブルＴ３、優先確認リストＬ１、優先確認外リストＬ２、ログ収集定義（ログ出力先定義）ＬＤ等が設けられる。
【００１７】
第１管理テーブルＴ１は、アプリプロセス毎のファイル識別子番号数（オープンしているファイルの数に相当）を保持するものである。アプリプロセスはいったんオープンしたファイルをクローズしないので、ファイル識別子番号数の増加により新たなファイルの出現を検出することができる。第２管理テーブルＴ２は、アプリプロセス毎で使用しているファイルのファイルパスを保持するとともに、収集対象であるか否かの判定結果を保持するものである。第３管理テーブルＴ３は、アプリプロセスのプロセス名と、そのアプリプロセスを監視の対象としている子監視プロセス名とを保持するものである。それぞれのテーブルの具体例については後述する。
【００１８】
優先確認リストＬ１は、子監視プロセスにおけるファイル名による簡易確認により、ログである可能性が高く、優先的に処理すべきアプリプロセスのプロセス名とファイルパス

を保持するものである。優先確認外リストＬ２は、子監視プロセスにおけるファイル名による簡易確認により、優先的に処理すべきとされなかったアプリプロセスのプロセス名とファイルパスを保持するものである。それぞれのリストの具体例については後述する。ログ収集定義ＬＤは、ログと判定されたアプリプロセスのプロセス名とファイルパスとが保持されるものであり、ログを収集する管理者により利用されるものである。

【００１９】

図４はログ収集システムの構成例を示す図である。図４（ａ）において、ログの収集対象となる情報処理装置１はアプリケーションサーバやネットワーク機器である。情報処理装置１にネットワーク２を介して接続されたログ収集サーバ３の管理のもと、ログ収集部ＬＣによりログを収集し、収集したログをログ収集サーバ３に送信する。ログ収集部ＬＣは、情報処理装置１の子監視プロセスにより自動的に生成されるログ収集定義ＬＤと管理者により設定された情報に基づき、ログの収集（ファイルの全内容または指定された内容の収集）を行う。

【００２０】

図４（ｂ）は、ログ収集サーバ３に代えてデータベースサーバ４を用いており、情報処理装置１のログ収集部ＬＣは、ログ収集定義と管理者により設定された情報に基づいてログを収集し、収集したログをデータベースサーバ４に送信して保存する。

【００２１】

図５は情報処理装置１のハードウェア構成例を示す図である。図５において、情報処理装置１は、バス１７を介して相互に接続されたＣＰＵ（Central Processing Unit）１１、ＲＯＭ（Read Only Memory）１２、ＲＡＭ（Random Access Memory）１３を備えている。また、情報処理装置１は、ＨＤＤ（Hard Disk Drive）／ＳＳＤ（Solid State Drive）１４、接続Ｉ／Ｆ（Interface）１５、通信Ｉ／Ｆ１６を備えている。ＣＰＵ１１は、ＲＡＭ１３をワークエリアとしてＲＯＭ１２またはＨＤＤ／ＳＳＤ１４等に格納されたプログラムを実行することで、情報処理装置１の動作を統括的に制御する。接続Ｉ／Ｆ１５は、情報処理装置１に接続される機器とのインタフェースである。通信Ｉ／Ｆ１６は、ネットワークを介して他の情報処理装置と通信を行うためのインタフェースである。

【００２２】

図３で説明した情報処理装置１の機能は、ＣＰＵ１１において所定のプログラムが実行されることで実現される。プログラムは、記録媒体を経由して取得されるものでもよいし、ネットワークを経由して取得されるものでもよいし、ＲＯＭ組込でもよい。処理に際して参照・更新されるデータは、ＲＡＭ１３またはＨＤＤ／ＳＳＤ１４に保持される。

【００２３】

＜動作＞

以下、フローチャートに沿って上記の実施形態の動作について説明する。また、フローチャートによる一般的な処理の他に、具体例についても併せて説明する。具体例については、親監視プロセスＰＰの起動時に、アプリケーション１とアプリケーション２の２つが実行中であり、それらに対応する子監視プロセスＣＰ１、ＣＰ２における一巡目の処理を終了した後に、アプリケーション３が新たに実行中になるものとしている。なお、各アプリケーションは次のようなファイルを使用するものとしている。

・アプリケーション１（プロセス１）
　出力ファイル名１：setting.conf（ファイル内に日時情報を含まない）
　出力ファイル名２：error.log（ファイル内に日時情報を含む）
・アプリケーション２（プロセス２）
　出力ファイル名１：errlog（ファイル内に日時情報を含む）
　出力ファイル名２：infolog（通常は出力されないが何らかの操作をすると出力されるファイル。ファイル内に日時情報を含む）
・アプリケーション３（プロセス３）
　出力ファイル名１：エラーログ.txt（ファイル内に日時情報を含まない）

【００２４】

［親監視プロセスの処理］
図６は親監視プロセスＰＰの処理例を示すフローチャートである。図６において、親監視プロセスＰＰは、管理者からの指示や自動起動設定に基づく指示等に基づいて起動されると、処理を開始する。この時点では、子監視プロセスはまだ存在しない。
【００２５】
親監視プロセスＰＰは、情報処理装置１のＯＳ（Operating System）からプロセス一覧を取得し、アプリプロセスを検出する（ステップＳ１０１）。図７はプロセス一覧の例を示しており、現在実行中のプロセス名と種別とが含まれており、種別からアプリプロセスを識別することができる。ここでは、前述したように、親監視プロセスＰＰの起動時に、アプリケーション１とアプリケーション２の２つが実行中であるものとして、それらのプロセス１とプロセス２の２つがアプリプロセスとして検出されるものとする。なお、プロセス一覧の各プロセス（プロセス名）にはプロセスＩＤが付されており、プロセスＩＤからプロセス情報をＯＳから取得することができる。プロセス情報には、そのプロセスが使用しているファイル一覧が含まれており、ファイル一覧からファイル識別子とファイルパスとを取得することができる。
【００２６】
図６に戻り、親監視プロセスＰＰは、検出したアプリプロセスにつき、第３管理テーブルＴ３を参照して、第３管理テーブルＴ３にない新規のアプリプロセスがあるか否か判定する（ステップＳ１０２）。新規のアプリプロセスがないと判定した場合（ステップＳ１０２のＮＯ）、親監視プロセスＰＰは、アプリプロセスの検出の処理（ステップＳ１０１）に戻る。図７に示した２つのプロセス１とプロセス２がアプリプロセスとして検出された場合、この時点（起動直後）での各管理テーブルＴ１～Ｔ３は図８に示すようにブランクであるため、２つのアプリプロセスとも新規であると判定される。
【００２７】
図６に戻り、親監視プロセスＰＰは、新規のアプリプロセスがあると判定した場合（ステップＳ１０２のＹＥＳ）、第３管理テーブルＴ３にないアプリプロセス（複数の場合はそれらの個々）に対応する子監視プロセスを起動する。そして、それらの子監視プロセスに各アプリプロセスの監視依頼を行う（ステップＳ１０３）。図９は、具体例に対応して、親監視プロセスＰＰが子監視プロセスＣＰ１、ＣＰ２を起動し、アプリプロセス１（プロセス１）とアプリプロセス２（プロセス２）の確認（監視）を子監視プロセスＣＰ１と子監視プロセスＣＰ２にそれぞれ依頼した状態を示している。
【００２８】
図６に戻り、親監視プロセスＰＰは、アプリプロセス名と、監視を担当する子監視プロセス名を第３管理テーブルＴ３に書き込み（ステップＳ１０４）、アプリプロセスの検出の処理（ステップＳ１０１）に戻る。図１０は、具体例に対応して、アプリプロセス１、２と、担当する子監視プロセスＣＰ１、ＣＰ２の情報を第３管理テーブルＴ３に書き込んだ状態を示している。
【００２９】
［子監視プロセスの処理］
図１１は子監視プロセス（ＣＰ１、ＣＰ２、・・）の処理例を示すフローチャートである。図１１において、親監視プロセスＰＰより起動された子監視プロセスは、以下の処理を開始する。
【００３０】
子監視プロセスは、第３管理テーブルＴ３から自己の監視対象のアプリプロセスを１つ選択し（ステップＳ２０１）、そのアプリプロセスのプロセス情報から現在のファイル識別子番号数を取得する（ステップＳ２０２）。具体例に対応して、図１０に示した第３管理テーブルＴ３から、子監視プロセスＣＰ１は、プロセス１を担当すると認識し、図７に示したプロセス１のプロセス情報における２つのファイル識別子に対応したファイル識別子番号数「２」を取得する。同様に、子監視プロセスＣＰ２は、プロセス２を担当すると認識し、プロセス情報における２つのファイル識別子のうち、infologについてはカウン

トされないため、ファイル識別子番号数「１」を取得する。

【００３１】

図１１に戻り、子監視プロセスは、取得した現在のファイル識別子番号数につき、第１管理テーブルＴ１のファイル識別子番号数から増加したか否か判定する（ステップＳ２０３）。ファイル識別子番号数に変更がない場合（ステップＳ２０３のＮＯ）、監視対象のアプリプロセスについて全て確認したか否かの判断（ステップＳ２０９）に移行する。

【００３２】

ファイル識別子番号数が増加したと判定した場合（ステップＳ２０３のＹＥＳ）、子監視プロセスは、第１管理テーブルＴ１にファイル識別子番号数を書き込む（ステップＳ２０４）。図１２の第１管理テーブルＴ１には、具体例に対応して、子監視プロセスＣＰ１と子監視プロセスＣＰ２がそれぞれファイル識別子番号数を書き込んだ状態を示している。

【００３３】

図１１に戻り、子監視プロセスは、監視対象のアプリプロセスのプロセス情報からファイル一覧を取得し（ステップＳ２０５）、ファイル名による簡易確認を行う（ステップＳ２０６）。

【００３４】

図１３はファイル名による簡易確認（図１１のステップＳ２０６）の処理例を示すフローチャートである。図１３において、子監視プロセスは、ファイル一覧からファイルを１つ選択し（ステップＳ２１１）、第２管理テーブルＴ２から確認対象ファイルであるか否か判定する（ステップＳ２１２）。すなわち、第２管理テーブルＴ２に格納されているファイルは確認済として、監視対象ではないとし、格納されていないファイルを監視対象と判定する。具体例において、子監視プロセスＣＰ１、ＣＰ２がこの時点で参照する第２管理テーブルＴ２は、図１２に示すようにブランクとなっているため、全て確認対象ファイルであると判定する。

【００３５】

図１３に戻り、確認対象ファイルであると判定した場合（ステップＳ２１２のＹＥＳ）、子監視プロセスは、ファイル名（拡張子を含む）に文字列「log」「ログ」が含まれるか否か判定する（ステップＳ２１３）。含まれると判定した場合（ステップＳ２１３のＹＥＳ）、子監視プロセスは、優先確認リストＬ１へプロセス名とファイルパスを格納する（ステップＳ２１４）。含まれないと判定した場合（ステップＳ２１３のＮＯ）、子監視プロセスは、優先確認外リストＬ２へプロセス名とファイルパス格納する（ステップＳ２１５）。具体例において、子監視プロセスＣＰ１は、図７に示したプロセス１のファイル一覧のsetting.confには文字列「log」「ログ」が含まれていないと判定し、errot.logには文字列「log」「ログ」が含まれると判定する。図１４は、優先確認リストＬ１にerrot.logのファイルパスが格納され、優先確認外リストＬ２にsetting.confのファイルパスが格納された状態を示している。

【００３６】

図１３に戻り、確認対象ファイルでないと判定した場合（ステップＳ２１２のＮＯ）と優先確認リストＬ１または優先確認外リストＬ２に格納（ステップＳ２１４、Ｓ２１５）の後、子監視プロセスは、ファイル一覧をすべて確認したか否か判定する（ステップＳ２１６）。ファイル一覧のすべてを確認していないと判定した場合（ステップＳ２１６のＮＯ）はファイルの選択（ステップＳ２１１）に戻り、すべてを確認したと判定した場合（ステップＳ２１６のＹＥＳ）は処理を終了する。

【００３７】

このようなファイル名による簡易確認のアルゴリズムにより、確認対象のファイルを優先確認リストＬ１と優先確認外リストＬ２に振り分けることができる。

【００３８】

図１１に戻り、子監視プロセスは、続いて、ファイル内容による詳細確認を行う（ステップＳ２０７）。

【００３９】
図１５はファイル内容による詳細確認（図１１のステップＳ２０７）の処理例を示すフローチャートである。図１５において、子監視プロセスは、優先確認リストＬ１から１つのファイルを選択し（ステップＳ２２１）、ファイルの内容を解析して、ファイルの内容がログか確認を行う（ステップＳ２２２）。例えば、ファイルの先頭行＋１行目（２行目以降）からファイルの内容を確認し、日時を示す文字列が２行連続している箇所が２箇所以上存在するかを確認し、そのようになっていれば、ログと特定する。設定ファイルなどにも日時情報が含まれる場合があり、１箇所に存在するだけでは判定を誤る可能性があるが、ヘッダ行を除き、日時情報の連続性および繰り返し性を考慮することで判定の精度を高めている。
【００４０】
ログと確認した場合（ステップＳ２２２のＹＥＳ）、子監視プロセスは、ログ収集定義ＬＤと第２管理テーブルＴ２に情報を書き込む（ステップＳ２２３）。ログ収集定義ＬＤには、プロセス名とファイルパスを書き込む。第２管理テーブルＴ２には、プロセス名とファイルパスと収集対象：対象を書き込む。ログと確認しなかった場合（ステップＳ２２２のＮＯ）、子監視プロセスは、第２管理テーブルＴ２にプロセス名とファイルパスと収集対象：対象外を書き込む（ステップＳ２２４）。第２管理テーブルＴ２に書き込まれたファイルについては、収集対象（対象、対象外）の如何にかかわらず、次回以降の処理では確認は行われない。
【００４１】
図１６は、具体例における子監視プロセスＣＰ１が、図１４に示した優先確認リストＬ１に格納されたファイルerror.logがログであると確認し、プロセス名とファイルパスと収集対象：対象を第２管理テーブルＴ２に書き込んだ状態を示している。
【００４２】
図１５に戻り、子監視プロセスは、優先確認リストＬ１の全てのファイルについて確認したか否か判定し（ステップＳ２２５）、まだであれば次のファイルの選択（ステップＳ２２１）に戻り、完了していれば次の処理に移行する。
【００４３】
次いで、子監視プロセスは、優先確認外リストＬ２から１つのファイルを選択し（ステップＳ２２６）、ファイルの内容を解析して、ファイルの内容がログか確認を行う（ステップＳ２２７）。確認の手法は、優先確認リストＬ１についての場合と同様である。
【００４４】
ログと確認した場合（ステップＳ２２７のＹＥＳ）、子監視プロセスは、ログ収集定義ＬＤと第２管理テーブルＴ２に情報を書き込む（ステップＳ２２８）。ログ収集定義ＬＤには、プロセス名とファイルパスを書き込む。第２管理テーブルＴ２には、プロセス名とファイルパスと収集対象：対象を書き込む。
【００４５】
ログと確認しなかった場合（ステップＳ２２７のＮＯ）、子監視プロセスは、第２管理テーブルＴ２にプロセス名とファイルパスと収集対象：対象外を書き込む（ステップＳ２２９）。
【００４６】
図１７は、子監視プロセスＣＰ１が、優先確認外リストＬ２に格納されたファイルsetting.confがログでないと確認し、プロセス名とファイルパスと収集対象：対象外を第２管理テーブルＴ２に書き込んだ状態を示している。
【００４７】
図１５に戻り、子監視プロセスは、優先確認外リストＬ２の全てのファイルについて確認したか否か判定する（ステップＳ２３０）。全て完了していなければ、次のファイルの選択（ステップＳ２２６）に戻り、完了していれば、優先確認リストＬ１と優先確認外リストＬ２を削除し（ステップＳ２３１）、処理を終了する。
【００４８】

図１８は、具体例において、並行に動作する子監視プロセスＣＰ１、ＣＰ２により第２管理テーブルＴ２に書き込みが行われた状態を示している。

【００４９】

以上の処理により、一巡目の確認の処理が完了する。この際、ログである可能性が高いファイルから優先的に確認を行うため、迅速にログを特定することが可能である。また、ログである可能性が高くないファイルについても、優先順位は落とすものの、確認は確実に行うため、判定漏れを防止することができる。

【００５０】

図１１に戻り、複数の子監視プロセスが起動されている場合、簡易確認と詳細確認の一巡目の処理の終了により、２番目以降の子監視プロセスは、代表とする１番目の子監視プロセスに監視を引き継いで終了する（ステップＳ２０８）。すなわち、複数のアプリプロセスをグループ化し、それを１つの子監視プロセスで監視を行うようにする。なお、１番目の子監視プロセスに監視を引き継ぐのに代え、何らかの基準でその時点での代表的な子監視プロセスを定め、その子監視プロセスに引き継ぐようにしてもよい。引き継ぎは、第３管理テーブルＴ３において、終了する側の子監視プロセス名を引き継ぎ先の子監視プロセス名に変更することで行う。

【００５１】

図１９は、具体例において、子監視プロセスＣＰ１、ＣＰ２が動作していた場合に、子監視プロセスＣＰ２がアプリプロセスＡＰ２についての監視を子監視プロセスＣＰ１に引き継ぎ、終了した状態を示している。図２０は引き継ぎ後のテーブルを示しており、第３管理テーブルＴ３のプロセス２についての子監視プロセス名は、以前（図１８）の子監視プロセス２から子監視プロセス１に変更されている。

【００５２】

新規のアプリプロセスの監視を開始した直後は処理負荷が高くなるため、各アプリプロセスに専用の子監視プロセスを割り当てて処理を行うことで、処理の効率化をはかっている。しかし、そのアプリプロセスについて一巡目の処理が終了した場合は、その後は新たに発生したファイルについてのみ処理を行えばよいため、専用の子監視プロセスを設ける必要性は低下する。そこで、アプリプロセスのグループ化を行い、１つの子監視プロセスに複数のアプリプロセスを対応させ、子監視プロセスの数を減らし、プロセスの多重度が高くなり過ぎないようにして、システムへの負荷が高くならないようにしている。

【００５３】

図１１に戻り、子監視プロセスは、監視対象のアプリプロセス（新規に起動された場合は１つ。グループ化された場合は複数）の全てを確認したか否か判定する（ステップＳ２０９）。そして、全てを確認していない場合は次のアプリプロセスについて最初の処理（ステップＳ２０１）から繰り返し、全てを確認した場合は、二巡目として最初のアプリプロセスについて最初の処理（ステップＳ２０１）から繰り返す。

【００５４】

図２１は、具体例において、アプリプロセスＡＰ１、ＡＰ２についての一巡目の処理が終了してグループ化が行われた後に、新たなアプリプロセスＡＰ３が実行中となることで、それに対応する子監視プロセスＣＰ３が新たに起動された状態を示している。

【００５５】

図２２は、子監視プロセスＣＰ３による確認が行われ、アプリプロセスＡＰ１、ＡＰ２、ＡＰ３についての確認結果が第２管理テーブルＴ２に格納された状態を示している。第２管理テーブルＴ２において収集対象が対象となっているファイルについては、プロセス名とファイルパスがログ収集定義ＬＤとして書き出されるため、ログ収集の管理者は、そのログ収集定義ＬＤに基づいて収集ツールへの設定を適切に行うことができる。

【００５６】

＜総括＞

以上説明したように、本実施形態によれば、特定の種別である可能性が高いファイルを優先的に種別判定の対象ファイルとすることができる。

【００５７】
　また、新たなアプリプロセスについては専用の子監視プロセスを割り当てることで、効率的に確認の処理を行うことができる。
【００５８】
　また、一巡目の処理を終了した場合に代表的な子監視プロセスに監視を引き継いで子監視プロセスを終了するため、監視プロセスの多重度が高くなり過ぎず、システムへの負荷を適切なものとすることができる。
【００５９】
　以上、好適な実施の形態により説明した。ここでは特定の具体例を示して説明したが、特許請求の範囲に定義された広範な趣旨および範囲から逸脱することなく、これら具体例に様々な修正および変更を加えることができることは明らかである。すなわち、具体例の詳細および添付の図面により限定されるものと解釈してはならない。
【００６０】
　以上の説明に関し、更に以下の項を開示する。
（付記１）
　実行中のプロセスが参照または更新する複数のファイルそれぞれの属性情報を記憶する記憶部を参照して、前記複数のファイルそれぞれについて、属性情報に特定の文字列が含まれるか否かの判定を行い、
　判定結果に基づき、属性情報に前記特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与するアルゴリズムを用いて、前記複数のファイルそれぞれに順位を付与し、
　前記複数のファイルそれぞれに含まれる情報を、付与された前記順位に従って順次解析し、
　解析結果に基づき、前記複数のファイルそれぞれのファイルの種別に関する判定を行う、
処理をコンピュータに実行させることを特徴とするファイル判定プログラム。
（付記２）
　前記属性情報は、ファイルの所在を示す情報である、
ことを特徴とする付記１に記載のファイル判定プログラム。
（付記３）
　前記種別に関する判定結果をファイルの属性情報に対応付けて記憶する、
ことを特徴とする付記１または２に記載のファイル判定プログラム。
（付記４）
　前記種別に関する判定結果を、さらに、実行中のプロセスに対応付けて記憶する、
ことを特徴とする付記３に記載のファイル判定プログラム。
（付記５）
　複数の前記プロセスそれぞれについて複数の監視プロセスが処理を行う第１のステップと、
　一巡目の処理を終了した監視プロセスの処理を他の監視プロセスが引き継いで以後の処理を行う第２のステップと、
を有することを特徴とする付記１乃至４のいずれか一項に記載のファイル判定プログラム。
（付記６）
　前記第２のステップは、担当する前記プロセスのいずれかのファイル識別子数の増加を検知した場合に、該プロセスについての処理を開始する、
ことを特徴とする付記５に記載のファイル判定プログラム。
（付記７）
　前記属性情報に特定の文字列が含まれるか否かの判定は、ファイル名にログであることを示す所定の文字列が含まれているか否かにより行う、
ことを特徴とする付記１乃至６のいずれか一項に記載のファイル判定プログラム。

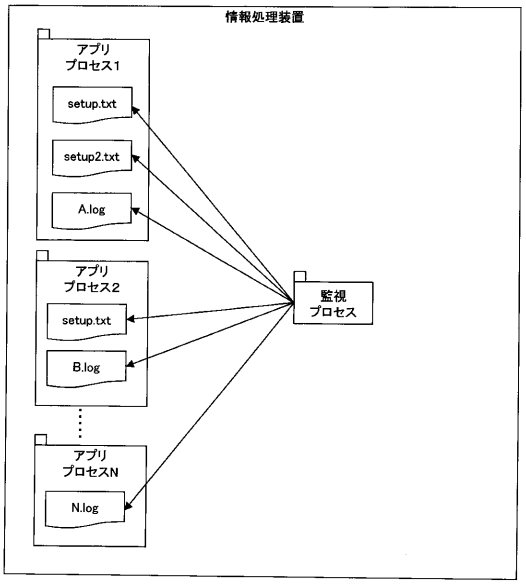
（付記８）
前記ファイルの種別に関する判定は、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かにより行う、
ことを特徴とする付記１乃至７のいずれか一項に記載のファイル判定プログラム。
（付記９）
実行中のプロセスが参照または更新する複数のファイルそれぞれの属性情報を記憶する記憶部を参照して、前記複数のファイルそれぞれについて、属性情報に特定の文字列が含まれるか否かの判定を行う手段と、
判定結果に基づき、属性情報に前記特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与するアルゴリズムを用いて、前記複数のファイルそれぞれに順位を付与する手段と、
前記複数のファイルそれぞれに含まれる情報を、付与された前記順位に従って順次解析する手段と、
解析結果に基づき、前記複数のファイルそれぞれのファイルの種別に関する判定を行う手段と、
を備えたことを特徴とするファイル判定装置。
（付記１０）
前記属性情報は、ファイルの所在を示す情報である、
ことを特徴とする付記９に記載のファイル判定装置。
（付記１１）
前記種別に関する判定結果をファイルの属性情報に対応付けて記憶する、
ことを特徴とする付記９または１０に記載のファイル判定装置。
（付記１２）
前記種別に関する判定結果を、さらに、実行中のプロセスに対応付けて記憶する、
ことを特徴とする付記１１に記載のファイル判定装置。
（付記１３）
複数の前記プロセスそれぞれについて複数の監視プロセスが処理を行う第１のステップと、
一巡目の処理を終了した監視プロセスの処理を他の監視プロセスが引き継いで以後の処理を行う第２のステップと、
を有することを特徴とする付記９乃至１２のいずれか一項に記載のファイル判定装置。
（付記１４）
前記第２のステップは、担当する前記プロセスのいずれかのファイル識別子数の増加を検知した場合に、該プロセスについての処理を開始する、
ことを特徴とする付記１３に記載のファイル判定装置。
（付記１５）
前記属性情報に特定の文字列が含まれるか否かの判定は、ファイル名にログであることを示す所定の文字列が含まれているか否かにより行う、
ことを特徴とする付記９乃至１４のいずれか一項に記載のファイル判定装置。
（付記１６）
前記ファイルの種別に関する判定は、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かにより行う、
ことを特徴とする付記９乃至１５のいずれか一項に記載のファイル判定装置。
（付記１７）
実行中のプロセスが参照または更新する複数のファイルそれぞれの属性情報を記憶する記憶部を参照して、前記複数のファイルそれぞれについて、属性情報に特定の文字列が含まれるか否かの判定を行い、
判定結果に基づき、属性情報に前記特定の文字列が含まれるファイルに対して、属性情報に前記特定の文字列が含まれないファイルよりも早い順位を付与するアルゴリズムを用いて、前記複数のファイルそれぞれに順位を付与し、

前記複数のファイルそれぞれに含まれる情報を、付与された前記順位に従って順次解析し、
解析結果に基づき、前記複数のファイルそれぞれのファイルの種別に関する判定を行う、
処理をコンピュータが実行することを特徴とするファイル判定方法。
（付記１８）
前記属性情報は、ファイルの所在を示す情報である、
ことを特徴とする付記１７に記載のファイル判定方法。
（付記１９）
前記種別に関する判定結果をファイルの属性情報に対応付けて記憶する、
ことを特徴とする付記１７または１８に記載のファイル判定方法。
（付記２０）
前記種別に関する判定結果を、さらに、実行中のプロセスに対応付けて記憶する、
ことを特徴とする付記１９に記載のファイル判定方法。
（付記２１）
複数の前記プロセスそれぞれについて複数の監視プロセスが処理を行う第１のステップと、
一巡目の処理を終了した監視プロセスの処理を他の監視プロセスが引き継いで以後の処理を行う第２のステップと、
を有することを特徴とする付記１７乃至２０のいずれか一項に記載のファイル判定方法。
（付記２２）
前記第２のステップは、担当する前記プロセスのいずれかのファイル識別子数の増加を検知した場合に、該プロセスについての処理を開始する、
ことを特徴とする付記２１に記載のファイル判定方法。
（付記２３）
前記属性情報に特定の文字列が含まれるか否かの判定は、ファイル名にログであることを示す所定の文字列が含まれているか否かにより行う、
ことを特徴とする付記１７乃至２２のいずれか一項に記載のファイル判定方法。
（付記２４）
前記ファイルの種別に関する判定は、該ファイルの２行目以降に日時を示す文字列が２行連続している箇所が２箇所以上存在するか否かにより行う、
ことを特徴とする付記１７乃至２３のいずれか一項に記載のファイル判定方法。
【符号の説明】
【００６１】

| | |
|---|---|
| １ | 情報処理装置 |
| ＰＰ | 親監視プロセス |
| Ｔ１ | 第１管理テーブル |
| Ｔ２ | 第２管理テーブル |
| Ｔ３ | 第３管理テーブル |
| ＡＰ１～ＡＰ３ | アプリプロセス |
| ＣＰ１～ＣＰ３ | 子監視プロセス |
| Ｌ１ | 優先確認リスト |
| Ｌ２ | 優先確認外リスト |
| ＬＤ | ログ収集定義 |
| ＬＣ | ログ収集部 |
| ２ | ネットワーク |
| ３ | ログ収集サーバ |
| ４ | データベースサーバ |

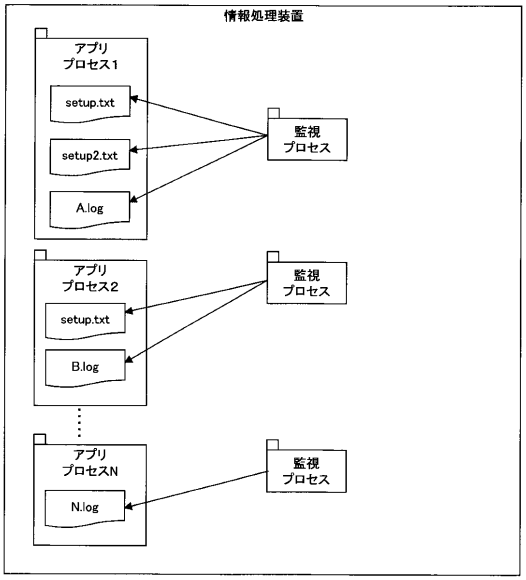
【図１】

従来におけるアプリケーションの出力するファイルの
監視の例を示す図（その1）



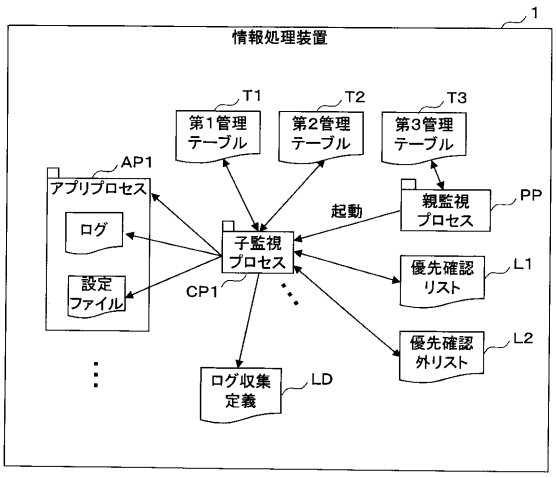

【図２】

従来におけるアプリケーションの出力するファイルの
監視の例を示す図（その2）



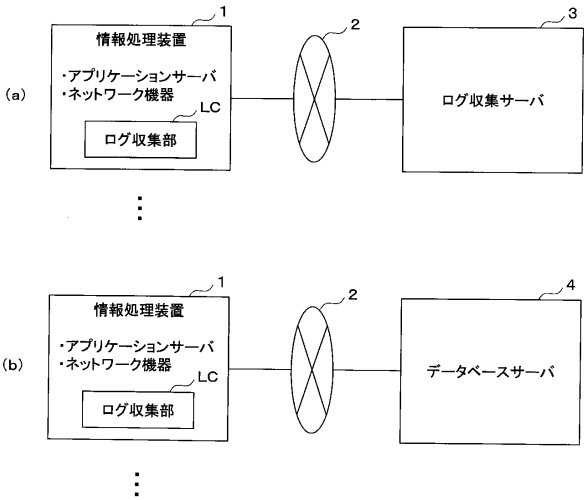

【図３】

情報処理装置の機能構成例を示す図



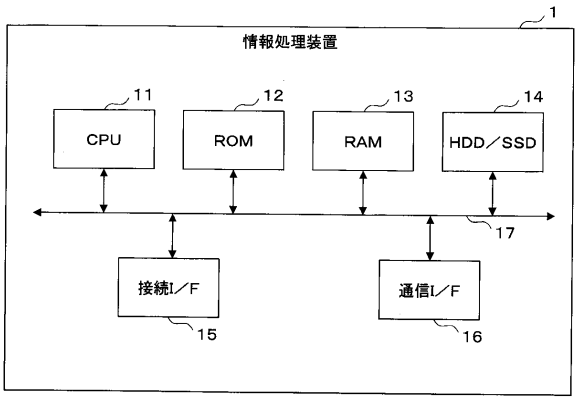

【図４】

ログ収集システムの構成例を示す図



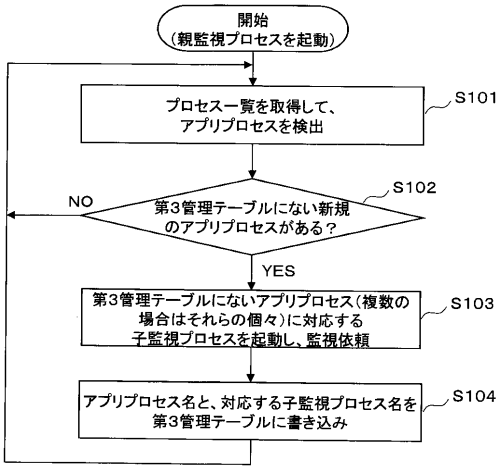
【図５】

情報処理装置のハードウェア構成例を示す図




【図６】

親監視プロセスの処理例を示すフローチャート




【図７】

プロセス一覧およびプロセス情報の例を示す図

プロセス一覧

| プロセスID | プロセス名 | 種別 |
|---|---|---|
| xxx | /opt/APP1/プロセス1 | アプリ |
| yyy | /opt/APP2/プロセス2 | アプリ |
| : | : | : |

プロセス情報

プロセスID:xxx

ファイル一覧

| ファイル識別子 | ファイルパス |
|---|---|
| xxx1 | /etc/opt/APP1/setting.conf |
| xxx2 | /var/opt/APP1/error.log |

プロセス情報

プロセスID:yyy

ファイル一覧

| ファイル識別子 | ファイルパス |
|---|---|
| yyy1 | /var/opt/APP2/errlog |
| yyy2 | /var/opt/APP2/infolog |

【図８】

管理テーブルの例を示す図

第1管理テーブル T1

| プロセス名 | ファイル識別子番号数 |
|---|---|
| - | - |

第2管理テーブル T2

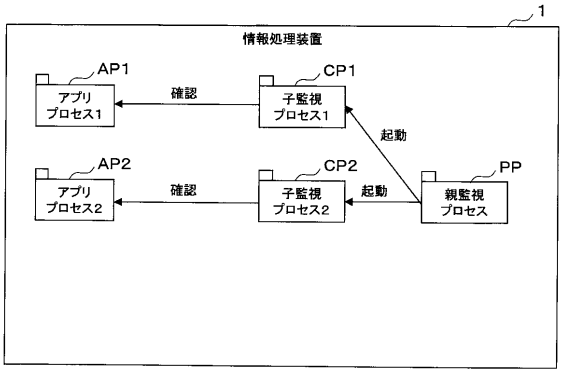
| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| - | - | - |

第3管理テーブル T3

| プロセス名 | 子監視プロセス名 |
|---|---|
| - | - |

【図９】

子監視プロセスを起動した状態の例を示す図




【図１０】

管理テーブルの例を示す図

第1管理テーブル T1

| プロセス名 | ファイル識別子番号数 |
|---|---|
| - | - |

第2管理テーブル T2

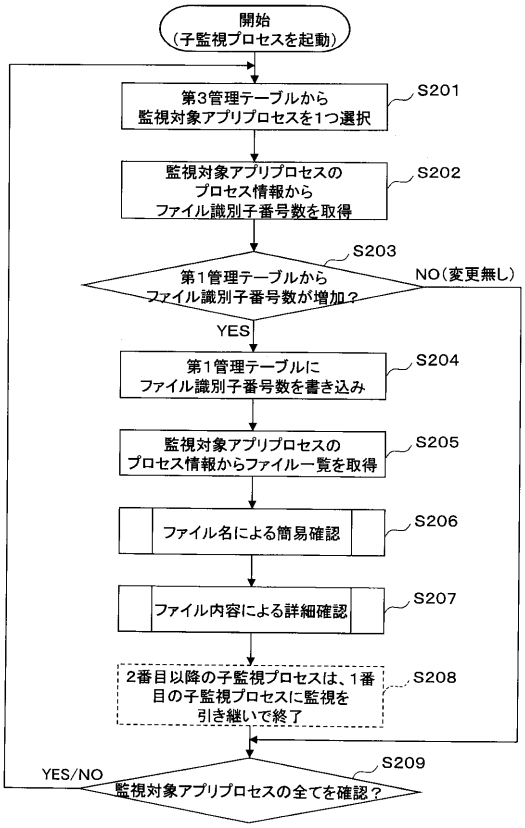
| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| - | - | - |

第3管理テーブル T3

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス2 |

【図１１】

子監視プロセスの処理例を示すフローチャート




【図１２】

管理テーブルの例を示す図

第1管理テーブル T1

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |

第2管理テーブル T2

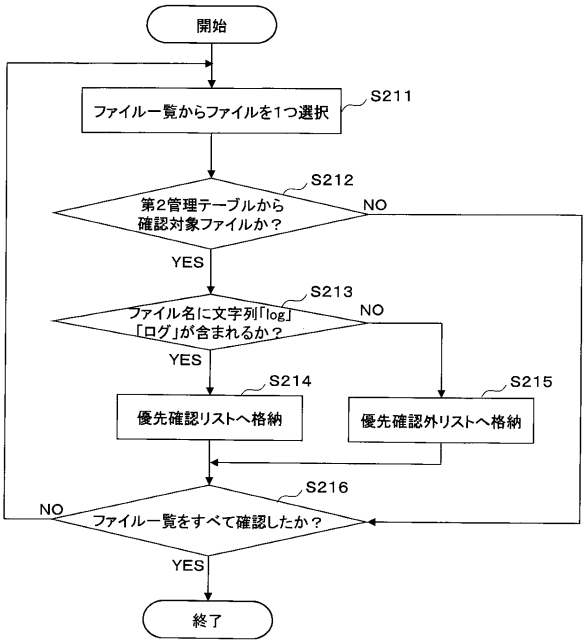
| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| - | - | - |

第3管理テーブル T3

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス2 |

【図１３】

ファイル名による簡易確認の処理例を示すフローチャート




【図１４】

優先確認リストおよび優先確認外リストの例を示す図

優先確認リスト

| プロセス名 | ファイルパス |
|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log |

L1

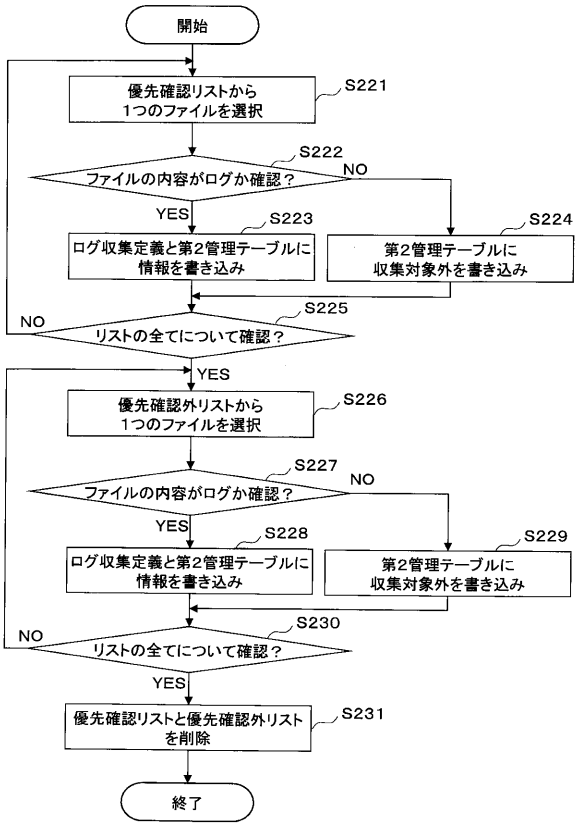
優先確認外リスト

| プロセス名 | ファイルパス |
|---|---|
| /opt/APP1/プロセス1 | /etc/opt/APP1/setting.conf |

L2

【図１５】

ファイル内容による詳細確認の処理例を示すフローチャート




【図１６】

管理テーブルの例を示す図

第1管理テーブル

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |

T1

第2管理テーブル

| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log | 対象 |

T2

第3管理テーブル

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス2 |

T3

【図１７】

管理テーブルの例を示す図

第1管理テーブル

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |

T1

第2管理テーブル

| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log | 対象 |
| | /etc/opt/APP1/setting.conf | 対象外 |

T2

第3管理テーブル

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス2 |

T3

【図１８】

管理テーブルの例を示す図

第1管理テーブル

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |

T1

第2管理テーブル

| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log | 対象 |
| | /etc/opt/APP1/setting.conf | 対象外 |
| /opt/APP2/プロセス2 | /var/opt/APP2/errlog | 対象 |
| | /var/opt/APP2/infolog | 対象 |

T2

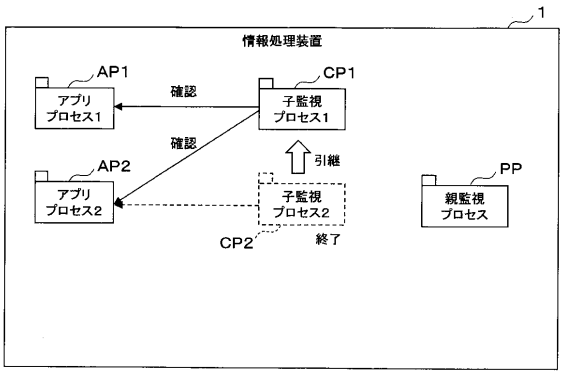
第3管理テーブル

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス2 |

T3

【図１９】

子監視プロセスを終了した状態の例を示す図




【図２０】

管理テーブルの例を示す図

第1管理テーブル

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |

T1

第2管理テーブル

| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log | 対象 |
| | /etc/opt/APP1/setting.conf | 対象外 |
| /opt/APP2/プロセス2 | /var/opt/APP2/errlog | 対象 |
| | /var/opt/APP2/infolog | 対象 |

T2

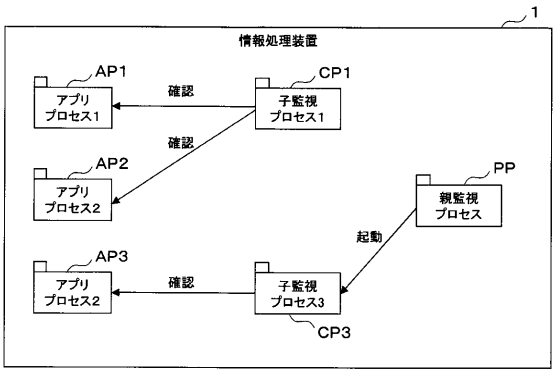
第3管理テーブル

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス1 |

T3

【図２１】

子監視プロセスを起動した状態の例を示す図




【図２２】

管理テーブルの例を示す図

第1管理テーブル

| プロセス名 | ファイル識別子番号数 |
|---|---|
| /opt/APP1/プロセス1 | 2 |
| /opt/APP2/プロセス2 | 1 |
| /opt/APP2/プロセス3 | 1 |

第2管理テーブル

| プロセス名 | ファイルパス | 収集対象 |
|---|---|---|
| /opt/APP1/プロセス1 | /var/opt/APP1/error.log | 対象 |
| | /etc/opt/APP1/setting.conf | 対象外 |
| /opt/APP2/プロセス2 | /var/opt/APP2/errlog | 対象 |
| | /var/opt/APP2/infolog | 対象 |
| /opt/APP3/プロセス3 | /var/opt/APP3/エラーログ.txt | 対象外 |

第3管理テーブル

| プロセス名 | 子監視プロセス名 |
|---|---|
| /opt/APP1/プロセス1 | 子監視プロセス1 |
| /opt/APP2/プロセス2 | 子監視プロセス1 |
| /opt/APP3/プロセス3 | 子監視プロセス3 |

フロントページの続き

(72)発明者　橋口　雅史
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　島野　淳
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

審査官　北元　健太

(56)参考文献　特開２００８－１９１８１３（ＪＰ，Ａ）
特開２０１３－１９１１８８（ＪＰ，Ａ）
特開２００８－２９３１７４（ＪＰ，Ａ）
特開２０１５－１３３０９８（ＪＰ，Ａ）
米国特許出願公開第２００４／０２１０８９８（ＵＳ，Ａ１）
特表２００１－５２５９６３（ＪＰ，Ａ）
特開２０１３－１９１０１２（ＪＰ，Ａ）
特開２００９－２３７７４９（ＪＰ，Ａ）
特開２００２－５６０２５（ＪＰ，Ａ）
特開２０１１－１４２６１４（ＪＰ，Ａ）
米国特許出願公開第２００３／０１５９０８８（ＵＳ，Ａ１）
韓国公開特許第１０－２０１４－００２０４１４（ＫＲ，Ａ）
特開２０１０－１０９９０７（ＪＰ，Ａ）
特開２００９－２１７３０６（ＪＰ，Ａ）
特開２００９－１９４７６５（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１１／２８　－　１１／３６
Ｇ０６Ｆ　１６／００　－　１６／９５８