

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6238983号
(P6238983)

(45) 発行日 平成29年11月29日(2017.11.29)

(24) 登録日 平成29年11月10日(2017.11.10)

(51) Int.Cl.

F I

G 0 6 F 12/00 (2006.01)

G 0 6 F 12/00 5 1 7

G 0 6 F 12/00 5 1 5 B

請求項の数 15 (全 19 頁)

(21) 出願番号	特願2015-521852 (P2015-521852)	(73) 特許権者	502303739
(86) (22) 出願日	平成25年7月12日(2013.7.12)		オラクル・インターナショナル・コーポレーション
(65) 公表番号	特表2015-525930 (P2015-525930A)		アメリカ合衆国カリフォルニア州94065レッドウッド・シティ、オラクル・パークウェイ500
(43) 公表日	平成27年9月7日(2015.9.7)		
(86) 国際出願番号	PCT/US2013/050323	(74) 代理人	110001195
(87) 国際公開番号	W02014/012022		特許業務法人深見特許事務所
(87) 国際公開日	平成26年1月16日(2014.1.16)	(72) 発明者	コクレイン、ネイル・ジェイムズ
審査請求日	平成28年7月8日(2016.7.8)		アメリカ合衆国、94114 カリフォルニア州、サン・フランシスコ、ナインティーンズ・ストリート、3822、ナンバー・2
(31) 優先権主張番号	13/547, 238		
(32) 優先日	平成24年7月12日(2012.7.12)		
(33) 優先権主張国	米国 (US)	審査官	後藤 彰
			最終頁に続く

(54) 【発明の名称】 開いているファイルの履歴閲覧

(57) 【特許請求の範囲】

【請求項 1】

コンピュータによって実現される方法であって、

プロセッサが、一組の関連するファイルのグラフをコンピュータディスプレイ上に表示するステップを備え、

前記グラフは、前記一組の関連するファイルの各々について、当該ファイルが開かれた時点を示す開始ポイント、当該ファイルが閉じられた時点を示す終了ポイント、および当該ファイルが保存された時点を示す1つ以上の中間チェックポイントを表わすための1つ以上のバーを有し、

前記方法はさらに、

前記プロセッサが、各ファイルの前記開始ポイント、前記終了ポイントおよび前記1つ以上の中間チェックポイントに対応する時点を含む履歴上の時点を含むタイムラインを、前記各ファイルに対応する1つ以上のバーと平行に表示するステップを備え、前記各ファイルは、前記タイムライン上の第1の時点において現在の状態の内容を有し、前記方法はさらに、

プロセッサが、前記第1の時点より前の特定の時点を示す入力を受取るステップと、

プロセッサが、前記一組の関連するファイルのファイル内容を備える環境を再作成するステップとを備え、

前記環境における前記ファイル内容は、前記特定の時点に関連付けられた履歴状態を有する、方法。

10

20

【請求項 2】

前記環境を表示するステップをさらに備え、前記環境を表示するステップは、前記特定の時点において開いていたファイル内容の状態の印を表示するステップを含む、請求項 1 に記載のコンピュータによって実現される方法。

【請求項 3】

プロセッサが、所望の時点およびバーの選択を受け付けるステップと、

プロセッサが、当該選択されたバーに対応するファイルの当該所望の時点における内容と、当該選択されたバーに対応するファイルの現在の内容との差異を特定できるように、当該選択されたバーに対応するファイルの内容を表示するステップとをさらに備える、請求項 1 または 2 に記載の方法。

10

【請求項 4】

前記一組の関連するファイルの前記グラフは、前記一組の関連するファイルにおける前記各ファイルに対応する 1 つ以上のバーを有する棒グラフであり、前記 1 つ以上のバーのうちの 1 つのバーの一方の側は、前記各ファイルの前記開始ポイントに対応する前記タイムライン上のある時点に対して位置決めされ、

前記バーのもう一方の側は、前記各ファイルの前記終了ポイントに対応する前記タイムライン上のある時点に対して位置決めされ、

前記バー内に 1 つ以上の印を表示するステップを備え、各印は、前記各ファイルの中間チェックポイントに対応する前記タイムライン上のある時点に対して位置決めされる、請求項 3 に記載のコンピュータによって実現される方法。

20

【請求項 5】

前記タイムラインは、水平な線として表示され、

前記 1 つ以上のバーは、履歴上のある時点において前記タイムラインと交差する垂直な線が、前記履歴上のある時点に対応する側における各バーまたは印とも交差するように、前記タイムラインに平行に表示される、請求項 3 に記載のコンピュータによって実現される方法。

【請求項 6】

前記タイムライン上の特定の時点を示す入力を受取るステップは、

前記タイムラインに直交する線を表示するステップをさらに備え、前記直交する線は、履歴スナップショットを時間で表し、前記タイムライン上の特定の時点を示す入力を受取るステップはさらに、

30

入力の受取りにตอบสนองして、前記直交する線が前記特定の時点において前記タイムラインと交差するように移動することを表示するステップを備える、請求項 3 に記載のコンピュータによって実現される方法。

【請求項 7】

前記グラフを表示するステップは、

特定の期間中に開いていたファイルの第 1 のリストを検索するステップと、

前記特定の期間内の任意の時刻において開いていた前記一組の関連するファイルにおける各ファイルごとに、

第 1 のファイルオープン時点として、前記各ファイルが開かれた前記特定の期間内の時刻を検索し、第 1 のファイルクローズ時点として、前記各ファイルが閉じられた対応する時刻を検索するステップと、

40

前記第 1 のファイルオープン時点に従って一方の側が位置決めされ、前記第 1 のファイルクローズ時点に従ってもう一方の側が位置決めされた第 1 のバーを表示するステップと、

第 1 のファイルセーブ時点として、前記各ファイルが保存された 1 つ以上の時刻を検索するステップとをさらに備え、第 1 のファイルセーブ時点は、前記第 1 のファイルオープン時点と前記第 1 のファイルクローズ時点との間にあり、前記グラフを表示するステップはさらに、

前記第 1 のバー上に 1 つ以上の印を表示するステップを備え、各印は、1 つ以上の前記

50

第 1 のファイルセーブ時点のうちの 1 つに対応する、請求項 1 に記載のコンピュータによって実現される方法。

【請求項 8】

前記グラフを表示するステップは、前記ファイルの第 1 のリストにおける各ファイルごとに、

第 2 のファイルオープン時点として、前記各ファイルが開かれた前記特定の期間内の時刻を検索し、第 2 のファイルクローズ時点として、前記各ファイルが閉じられた対応する時刻を検索するステップをさらに備え、

前記第 2 のファイルオープン時点は、前記第 1 のファイルクローズ時点の後であり、前記グラフを表示するステップはさらに、

前記第 2 のファイルオープン時点に従って一方の側が位置決めされ、前記第 2 のファイルクローズ時点に従ってもう一方の側が位置決めされた第 2 のバーを表示するステップと、

第 2 のファイルセーブ時点として、前記各ファイルが保存された 1 つ以上の時刻を検索するステップとをさらに備え、第 2 のファイルセーブ時点は、前記第 2 のファイルオープン時点と前記第 2 のファイルクローズ時点との間にあり、前記グラフを表示するステップはさらに、

前記第 2 のバー上に 1 つ以上の印を表示するステップを備え、各印は、1 つ以上の前記第 2 のファイルセーブ時点のうちの 1 つに対応する、請求項 7 に記載のコンピュータによって実現される方法。

【請求項 9】

前記ファイルの第 1 のリストを検索するステップは、時間ベースの索引に基づく、請求項 7 に記載のコンピュータによって実現される方法。

【請求項 10】

前記第 1 のファイルセーブ時点を検索するステップは、時間ベースの索引以外の表から前記第 1 のファイルセーブ時点を検索するステップをさらに備える、請求項 7 に記載のコンピュータによって実現される方法。

【請求項 11】

前記一組の関連するファイルは、共通のプロジェクトに属しており、

前記一組の関連するファイルにおける複数の前記ファイルは、共通のディレクトリ内にあり、

前記一組の関連するファイルの少なくとも 1 つのファイルは、前記共通のディレクトリの外側にある、請求項 1 から 10 のいずれか 1 項に記載のコンピュータによって実現される方法。

【請求項 12】

前記一組の関連するファイルは、統合開発環境内で管理されるソースファイルを備え、各ソースファイルは、コンピュータプログラミング言語に準拠するテキストを含む、請求項 1 から 10 のいずれか 1 項に記載のコンピュータによって実現される方法。

【請求項 13】

前記統合開発環境をさらに備え、

前記統合開発環境において 1 つ以上の開いているファイルを保存するためにファイルセーブ時点においてトリガを受取るステップと、

前記グラフにおいて前記ファイルセーブ時点を特定するためのラベルを受取るステップと、

前記保存された 1 つ以上の開いているファイルを前記ラベルに基づいて索引付けするために索引項目を作成するステップとをさらに備える、請求項 12 に記載のコンピュータによって実現される方法。

【請求項 14】

システムであって、

コンピュータ読取可能な媒体とプロセッサとを含むコンピュータを備え、前記コンピュ

10

20

30

40

50

ータは、コンピュータディスプレイに結合され、前記システムはさらに、
前記コンピュータ上で実行されるファイル管理アプリケーションを備え、
前記ファイル管理アプリケーションは、一組の関連するファイルのグラフを前記コンピュータディスプレイ上に表示するように構成され、

前記グラフは、前記一組の関連するファイルの各々について、当該ファイルが開かれた時点を示す開始ポイント、当該ファイルが閉じられた時点を示す終了ポイント、および当該ファイルが保存された時点を示す1つ以上の中間チェックポイントを表わすための1つ以上のバーを有し、

前記ファイル管理アプリケーションは、各ファイルの前記開始ポイント、前記終了ポイントおよび前記1つ以上の中間チェックポイントに対応する時点を有する履歴上の時点を含むタイムラインを、前記各ファイルに対応するバーと平行に表示するように構成され、

10

前記各ファイルは、前記タイムライン上の第1の時点において現在の状態の内容を有し、
前記ファイル管理アプリケーションはさらに、

前記第1の時点より前の特定の時点を示す入力を受取るように構成され、

前記一組の関連するファイルのファイル内容を備える環境を再生するように構成され、

前記環境における前記ファイル内容は、前記特定の時点に関連付けられた履歴状態を有する、システム。

【請求項15】

請求項1から13のいずれか1項に記載のコンピュータによって実現される方法を1つ以上のプロセッサに実現させるためのプログラム。

20

【発明の詳細な説明】

【背景技術】

【0001】

発明の背景

本明細書に記載されているアプローチは、一般に、コンピュータファイルの内容のバージョン管理に向けられる。電子エンティティは、互いに別々に格納および維持されるファイルの内容の集合体を備えてもよく、これらのファイルのうちのいくつかは、相互に依存し得る。このようなエンティティの一例は、ソフトウェアアプリケーションのためのソースコードであり得て、当該ソースコードは、複数のソースファイルに分散されている。別の例は、複数のチームまたは個人のためのタスクリストおよび期限を含むファイルに分散されたマスタスケジュールであってもよい。別の例は、本であってもよく、そこでは章は、異なるファイルに別々に維持されるが他の章内の場所との相互参照を含む。エンティティの変更は、複数のファイルの修正を必要としてもよく、全ての修正が完了するまでエンティティは一貫した状態であることができない。

30

【発明の概要】

【発明が解決しようとする課題】

【0002】

ファイルなどの独立して更新されるコンポーネントに格納された内容からなる電子エンティティを開発または維持する際の一般的な問題は、履歴上のある時点におけるエンティティの一貫したスナップショットを取得することである。たとえ各コンポーネントが一連の状態を経て進展し得るとしても、全てのコンポーネントにわたるエンティティの全体的な状態を視覚化できることは有用であろう。例えば、ソフトウェア開発プロジェクトは、コンパイルおよび連結されて実行可能なファイルを形成する複数のソースファイルを備え得る。1つのソースファイルが、異なるソースファイルに存在するコードによって呼び出される機能を定義し得る。当該機能を再定義するために当該1つのソースファイルを変更すると、当該機能を呼び出す異なるソースファイルにおけるコードは、対応する変更を必要とし得る。したがって、たとえこれらのファイルが独立して更新され得るものであっても、呼び出しコードを含むファイルは、機能を定義するファイルに対する依存性を有する。

40

【0003】

50

時間とともに、多くのコンポーネントファイルが開かれたり、編集されたり、閉じられたりしたであろう。往々にして、ユーザは、ファイルを取扱って、それらの変更を以前の状態に戻したいと判断することがあった。例えば、ユーザは、ここ数時間の作業に納得できず、以前のポイントからやり直したいと考えるのである。しかし、一組の関連するファイルを履歴上のある時点に戻すことは、時間がかかり、エラーが発生しやすい恐れがある。このプロセスに対してツールサポートを提供しようとする従前の試みは、一度に1つのファイルの変更を閲覧し、一度に1つのファイルの変更を戻すことのみを可能にした。また、別々のファイルであるが関連するファイルの変更を示すタイムラインを見る簡単な方法はなかった。

【0004】

10

ソースコード制御システムは、個々のファイルのバージョンの追跡に対してサポートを提供してもよく、これによりユーザはファイルを任意の追跡されたバージョンに回復させることができる。ユーザ要求時、ファイルの新たなバージョンが作成されてもよく、当該バージョンに名前が割り当てられ得る。また、いくつかのこのようなシステムは、関連するファイルの群の命名されたスナップショットを作成して、当該ファイルの群における各ファイルを命名されたスナップショットバージョンに巻き戻すことができるようにすることも可能にする。しかし、スナップショットバージョンに巻き戻すためには、ユーザは、スナップショットを使用する必要があるより前にスナップショットを作成するよう要求し、ユーザは、どの状態に巻き戻したいかを推測的に知ることができない。

【0005】

20

バックアップシステムは、通常は周期的な間隔でまたは要求時にファイルを格納し、そのため、当該システムは、システム全体の状態をバックアップが実行された時の状態に戻すために、格納されたファイルを用いることができる。バックアップされるファイルは、当該システム内の全てのファイルを含んでもよく、これらのファイルは、互いに関連付けられていないかもしれない。システム全体を回復させるためにどのバックアップを用いるべきであるか、またはどのバックアップから特定のファイルもしくはファイルの群を回復させるべきであるかを視覚的に判断することをサポートしようとする試みがなされてきた。アップル社のタイムマシンバックアップシステムは、まるでタイムマシンで飛行するようにフォルダを閲覧することを可能にするユーザインターフェースを提供する。任意の時点においてフォルダが出現することは、どのバックアップがどのファイルを含むかという点に関する手がかりを提供する。しかし、このアプローチは、特定のファイルの必要なバージョンをユーザが特定することを助けるが、相互に依存したファイルがファイルシステムまたは分散ファイルシステムの中のどこに存在し得ても、相互に依存したファイル同士の関係の理解に対してはサポートを提供しない。

30

【0006】

ユーザが推測的に一貫したスナップショットを作成および命名することに依拠する必要なく、関連するファイルの群に注目して、一組の関連するファイルをどの時点のスナップショットに回復させるべきかを判断するための使いやすい、直感的な方法が必要である。

【課題を解決するための手段】

【0007】

40

発明の概要

本発明のさまざまな実施例に係るシステムおよび方法は、コレクション内の全てのファイルについてチェックポイント状態を推測的に作成することなく、ユーザがファイル内容の状態を閲覧して、任意の時点において存在していた通りにファイル内容の状態を戻すことを可能にするユーザインターフェースを提供することによって、ファイル管理を実行する既存のアプリケーションにおける欠陥に対処することができる。また、当該ユーザインターフェースは、より大きなエンティティの状態を備えるファイル状態同士の関係の視覚的表示も提供し得る。例えば、ある実施例では、一組の関連するファイルのグラフがコンピュータディスプレイ上にプロセッサによって表示され得て、当該グラフは、一組の関連するファイルの各ファイルごとに、開始ポイント、終了ポイントおよび1つ以上の中間チ

50

チェックポイントを示す。開始ポイント、終了ポイントおよび1つ以上の中間チェックポイントは各々、ある時点を表す。各時点は、ファイルのファイル内容の状態に関連付けられる。例えば、現在の時刻である「現在」は、ファイルの現在の状態に関連付けられ得る。プロセッサは、現在の時刻より前の特定の時点を示す入力を受取り得る。プロセッサは、受取られた特定の時点における一組の関連するファイルの履歴ファイル内容を備える環境を作成し得る。

【図面の簡単な説明】

【0008】

【図1】本発明の実施例に係る、ファイル状態を経時的に示す棒グラフである。

【図2】本発明の実施例に係る、ファイルの各々の状態を回復させる時刻を示すためのスナップショット線を示す図である。

10

【図3a】本発明の実施例に係る履歴タイムライン上の時点ラベルを示す図である。

【図3b】本発明の実施例に係る履歴タイムライン上の時点ラベルを示す図である。

【図3c】本発明の実施例に係る履歴タイムライン上の時点ラベルを示す図である。

【図4】本発明の実施例に係る履歴上のある時点の選択を示す図である。

【図5】本発明の実施例に係る特定の時点におけるファイルの状態の閲覧を示す図である。

。

【図6】実施例に係る、本明細書に記載されている技術に関連付けられる高レベルのステップを示す図である。

【図7】本発明の実施例に係るファイル状態を再作成するために用いられるデータストアを示す図である。

20

【図8】本発明の実施例に係るファイル状態グラフを生成するフロー図である。

【図9】本発明の実施例に係る、特定の時点において開いていたファイルの環境を再作成するフロー図である。

【図10】実施例が実現され得るコンピュータシステムのブロック図である。

【発明を実施するための形態】

【0009】

発明の詳細な説明

多くのソフトウェアアプリケーションは、特注のファイル管理を実行するコンポーネントを含んでいる。1つのこのようなソフトウェアアプリケーションの一例は、統合開発環境(integrated development environment: IDE)である。IDEは、承認、修正、コンパイル、導入およびデバッグソフトウェアを含み得るソフトウェア開発の多くの局面をサポートするツールをコンピュータプログラマに提供する。本明細書に開示されている本発明の実施例は、一例としてIDEによって実行されるものとして記載されているが、当該アプローチは、複数の相互に依存している電子部品の内容のバージョン制御を実行するいかなるソフトウェアアプリケーションにも使用可能である。

30

【0010】

プロジェクトは、時として、複数のコンピュータファイルにわたって格納される。これらのプロジェクトを取扱うユーザは、相互に依存し得るこれらのファイルに変更を加える。1つのファイルに変更を加えることにより、別のファイルを変更しなければならなくなり得る。ユーザは、プロジェクトファイルを管理するためにIDEに依拠し得て、当該IDEは、時間とともに多くのファイルを開いたり、編集したり、閉じたりするために用いられてきたであろう。往々にして、ユーザは、ファイルに変更を加えた後で、それらの変更を以前の状態に戻そうと判断することがあった。例えば、ユーザは、ここ数時間の作業に納得できず、以前のポイントからやり直したいと考えるのである。1つのプロジェクトファイルの状態を回復させることにより、他の相互に依存するファイルの状態を回復させなければならなくなり得る。

40

【0011】

本明細書に記載されているアプローチは、時間とともに開かれたり、編集されたり、閉じられたりしたファイルの履歴を示す新たな種類のグラフを提供する。一実施例では、グ

50

ラフが生成され、当該グラフは、「現在」から時間を遡るように延びるグラフの底部に沿ってタイムラインを示す。開かれた各ファイルは、ファイルが開かれた時刻（開始ポイント）とファイルが閉じられた時刻（終了ポイント）との間に時間を遡るように延びる水平なバーによって表され、開始ポイントおよび終了ポイントは、タイムライン上に対応するラベル付きの時点を示している。他の実施例では、他の種類のグラフが用いられてもよく、当該グラフ内の各オブジェクトは、ファイルを表し、そのオープン、クローズおよび中間セーブの時刻を示す。

【0012】

図1は、本発明の実施例に係る、ファイル状態を経時的に示す棒グラフである。当該グラフは、Y軸の部分が別個のファイルを表し、X軸がタイムラインを表すように描かれている。棒グラフのバーは、タイムラインに平行に描かれ得る。グラフ内の各バーは、単一のファイルに対応し、Y軸の同一の部分を占める複数のバーは、複数回開かれたり閉じられたりした同一のファイルを表し得る。より具体的には、Y軸上で同一の高さにあるバーは、同一のファイルを表し得て、異なる高さにあるバーは、異なるファイルを表し得る。当該グラフは、特定の期間内に使用された全てのファイルを示し得る。例示的なグラフ100によって表される期間では、5つのファイルが開かれたり閉じられたりした。当該ファイルは名前を付けられ、それらの幅は、開いていた時間の長さを示す。例えば、バー110は、「File-A.txt」と命名されたファイルを表している。現在の時刻（現在）が原点であり、履歴時間は原点の左側に表されている（古）。バー内の点線印は、対応するファイルが保存された時を示している（すなわち、点線は、チェックポイントが生じた時刻に対応してX軸に対して位置決めされる）。ユーザは、任意の時点において任意のバーの上にマウスを乗せ得て、その時点における対応するファイルの状態を表示する追加のUIを閲覧することができる。

【0013】

図2は、本発明の実施例に係る、ファイルの各々の状態を調べる時刻を指定するためのスナップショット線を示す。グラフ200は、スナップショット線210を、同一の特定の時点における全てのバーと交差する、タイムラインに直交する垂直な破線として示している。全てのファイルの状態を調べて、恐らくは回復させる時刻を選択するために、垂直な線は、タイムラインに沿って左または右にドラッグされ得る。ある実施例では、ユーザが、キーボードまたは時刻を選択するためのその他の機構を介して時刻を入力し得て、IDEが、選択された時点におけるファイルと交差するようにスナップショット線の位置を更新し得る。

【0014】

図3a、図3bおよび図3cは、本発明の実施例に係る履歴タイムライン上の時点ラベルを示す。図3aは、タイムスタンプでラベル付けされる特定のポイントを履歴タイムライン上に示す。この例では、ファイルオープンイベント、ファイルセーブイベントおよびファイルクローズイベントが起こった時刻が、ラベル付きの時点として選択される。これらのタイムスタンプは、ファイルイベントに関連付けて格納され得る。この図では、最も古い時点は2012年4月21日であり、最も最近の時点は2012年5月31日7:57pmである。タイムラインは、一定の比率に応じている場合もあれば、この例に示されるように、時点は一定の比率に応じていない場合もある。この例では、最近の時点は、日付、時間および分を含む時間でラベル付けされているが、古い時点は、単に日および時間または日のみを用いてラベル付けされ得る。この非線形のタイムスケーリングの背景にある考え方は、最も古い時点は最も関心を引きにくいいため、より古いチェックポイントほど表示され得る数が少ない、というものである。また、当該非線形のスケーリングにより、線形のスケーリングで可能であろう期間よりも広い期間を表示することが可能になる。

【0015】

図3bは、ユーザにとって意味を持つテキストラベルの使用を示す。例えば、ファイルがソフトウェア開発プロジェクトの一部であり、プロジェクトライフサイクルが特定のライフサイクルマイルストーンにおいて全てのファイルをチェックポイント化する（状態を保

10

20

30

40

50

存する)ことを求める場合には、チェックポイントが生じた日付および時刻よりもマイルストンの名前が意味があるであろう。図3bに表示されている例示的なラベルとしては、バージョンA、点検チェックポイント、ユニットテストチェックポイント、バージョンシステムテストおよびバージョンテストが挙げられる。イベントが起こると、ユーザは、当該イベントを関連付けるためにラベルを入力または選択するよう促され得る。また、時点ラベルは、タイムライン上のラベルを選択してそれを新たなテキストラベルで改名することによっても変更され得る。

【0016】

図3cは、経時的な変更の数および分布に基づいて選択される相対時刻ラベルの使用を示す。例えば、最も右側の時点「-1時間」は、現在の時刻の1時間前を表す。どの相対時点を表示すべきかを判断するためにファイルオープンイベント、ファイルセーブイベントおよびファイルクローズイベントの実際の時刻が用いられ得るが、ラベル付きの時点は、ファイルイベントが起こった時刻と一致している必要はない。ある実施例では、グラフ内に垂直な線によって表される開始ポイント、終了ポイントおよび中間保存ポイントは、各々、対応するイベントが起こった時刻でさらにラベル付けされてもよい。別の実施例では、ファイルイベントが起こった実際の時刻は、カーソルがバー内の垂直な線を指すかまたは選択した時に表示されてもよい。

【0017】

ユーザは、履歴上の任意の時点を選択し、選択された時点において個々のファイルに格納されていた内容の状態を調べ得る。図4は、本発明の実施例に係る履歴上のある時点の選択を示す。ユーザは、ファイルの状態が所望である時刻を示すために、グラフを横切るように垂直な線をスライドさせ得る。

【0018】

図5は、本発明の実施例に係る特定の時点におけるファイルの状態の閲覧を示す。ある実施例では、ファイルおよび所望の時点は、カーソル500の位置に基づいて選択され得る。例えば、ファイルは、当該ファイルを表すバー上にカーソルを位置決めすることによって選択され得る。同様に、所望の時点は、タイムラインに対するカーソルの位置に基づいて選択され得る。カーソル位置は、カーソルを乗せることによって、またはマウスボタンをクリックすることによって決定され得る。代替的な実施例では、スナップショット線の場所が、調べられる任意のファイルの時点を決してもよい。ある実施例では、ファイルは、ファイル名をテキストボックスにタイプすることによって、またはドロップダウンもしくはポップアップメニューからファイル名を選択することによって選択され得る。

【0019】

所望の時点およびファイルが選択されると、選択されたファイルにおける内容の状態が、選択された時点において表示され得る。図5における例は、対象のファイルであるFile_C.txtを表すバーの上にマウスが乗っていることを示している。選択された時点におけるファイル内容は、ポップアップウィンドウ510に提示され得る。ある実施例では、ファイルの内容全体が、ファイルの現在の状態から独立して、当該時点において表示され得る。別の実施例では、ファイル内容の表示は、履歴バージョンと現在のバージョンとの内容の結合を含み、履歴バージョンと現在のバージョンとの間で内容がいかに変化したかを示し得る。ファイルのバージョン間の違いを特定する方法は、例えば、変化した要素を特定するための色、下線および/または取消線記号の使用を含み得る。図5では、網掛け領域が、選択された履歴上の時点以降に変更された内容を示し得る。さらに別の実施例では、履歴状態と現在の状態との違いのみが表示され、履歴バージョンと現在のバージョンとの両方のバージョンに共通する内容は表示されなくてもよい。

【0020】

ユーザは、正確な時点が選択されたことを検証するために、ある時点における特定のファイルの状態を点検し得る。検証されると、ユーザは、全てのファイルが、選択された時点において有していた状態に戻る環境を作成するよう要求し得る。例えば、図4を参照して、ユーザがスナップショット線の位置によって選択された時点に基づいて環境を作成す

10

20

30

40

50

るよう要求した場合、当該環境は、開かれた時のFile-A.txt、初めて保存された後のFile-B.txt、および閉じられる直前のFile-C.txtのバージョンを含むであろう。当該環境には、File-D.txtおよびFile-E.txtのファイルは存在しないであろう。なぜなら、それらのファイルは、5 / 2 9 / 1 2 1 3 : 0 0 から 5 / 2 0 / 1 2 1 0 : 1 7 までの間は開いていなかったからである。

【 0 0 2 1 】

ある実施例では、チェックポイント同士の間、例えばオープンとセーブとの間またはセーブ同士の間にはファイルに加えられの変更は、状態に反映されないであろう。各ファイルの状態は、その最後のオープンまたはセーブの時点の状態であろう。1つ以上のファイルの内容の保存は、ユーザによって引起される（ユーザ起動である）場合もあれば、IDE 10におけるタイマによって引起される場合もある。ある実施例では、各バーにおける点線の保存される線が、ユーザ起動の保存を表し得る。ユーザ起動の保存は、個々のファイルを保存する場合もあれば、全てのファイルを同一の時点で保存する場合もある。たとえIDEが複数のファイルを保存するために逐次的にファイルを反復し得て、そのため、ファイル全体にわたる実際の保存時刻が異なり得るとしても、IDEは、コレクション内の各ファイルについての保存ポイントとして共通の時点割り当て得る。また、IDEは、ユーザ15起動の保存同士の間にはファイルのスナップショットを自動的に保存し得る。自動的なスナップショットは、同一の点線を用いて、何らかの他の記号を用いてバー内に示される場合もあれば、全く示されない場合もある。そして、特定の時点におけるファイルの状態は、ユーザが保存を起動したかIDEが保存を起動したかの最後のチェックポイント以降の20ファイルの状態であり得る。

【 0 0 2 2 】

図6は、実施例に係る、本明細書に記載されている技術に関連付けられる高レベルのステップを示す。ステップ610において、特定の期間に関連付けられるファイル状態のグラフが生成され、表示される。ステップ620において、IDEは、所望の時点を示すユーザ25入力を受取る。所望の時点の受取りに応答して、所望の時点におけるタイムラインと交差するようにスナップショット線が再び引かれる。ステップ630において、所望の時点における状態のファイルを備える新たな環境が作成され得る。

【 0 0 2 3 】

これらの高レベルのステップを実現するために、グラフィング環境および履歴データ30のストアが用いられ得る。グラフィング環境は、突き詰めれば、グラフィカルコンピュータユーザインターフェースであり、任意の線およびテキストをその上に描くことができる機能を有している。データストアは、IDEによって開かれた全てのファイルのリストと、各々が開かれたり閉じられたりした時刻と、ファイルが開いていた間のさまざまな時点におけるファイルの状態とを含み得る。IDEは、ファイルのオープン、クローズおよびセーブを遮る制御機構を含み得る。これらのイベントのうちの1つが起これば、当該イベント35に関連付けられるデータが、当該ファイルのためのデータストアに入れられ得る。

【 0 0 2 4 】

図7は、本発明の実施例に係るファイル状態を再作成するために格納される例示的なデータを示す。イベントログ700は、ファイルオープンイベントおよびファイルクローズ40イベントの時間順のリストである。イベントログ700は、任意の時点においてどのファイルが開いていたかを判断するため、および、タイムライン上でいかに各ファイルをグラフ化するかを判断するために用いられ得る。本発明の実施例では、各ファイルの内容が保存された時刻を記録するために、イベントログ700から分離した別個のセーブイベントリスト710が用いられ得る。全てのファイルを一緒にチェックポイント化する上記の例では、代表的な時点が選択され、この同一の代表的な時刻を有するセーブイベントリストに全てのファイルが記録されるであろう。

【 0 0 2 5 】

ある実施例では、イベントログ700およびイベントセーブリスト710は、イベント50に関連付けられるテキストラベルを格納するための追加の列を含み得る。

【 0 0 2 6 】

保存された時の各ファイルの状態は、本明細書ではファイル状態データストア 720 と呼ばれるさらに別のデータストアに格納され得る。ファイルに対する修正が保存されると、当該時点におけるファイル状態を再構築するために必要な情報がファイル状態データストア 720 に保存される。ある実施例では、ファイル内容のコピー全体が格納され得る。代替的な実施例では、最後のチェックポイント以降に加えられた変更がファイル状態データストア 720 に書込まれてもよい。ある実施例では、イベントログ 700 およびセーブイベントリスト 710 は、時間でソートされた単一のリストに併合され得る。

【 0 0 2 7 】

ユーザが関連するファイルのグラフの閲覧を要求すると、ファイルの状態を表示することになる履歴上の期間が決定される。履歴上の期間は、グラフのタイムラインによって表される、現在の時刻から遡る時間量である。履歴上の期間は、ユーザからの入力として受取られる場合もあれば、IDE 内で構成される場合もあれば、少なくともイベントログに格納されるイベントの数および当該イベントを含む期間を考慮する規則に基づいて動的に決定される場合もある。図 8 は、本発明の実施例に係るファイル状態グラフの生成を示すフロー図である。イベントログは、グラフが表す期間の開始後に、第 1 の時刻項目から始まって、時系列順に読取られ得る。例えば、図 7 に示される例示的なデータに基づいて、現在の時刻が 3 : 30 pm であり、グラフが同日早くの 10 : 00 am に開始するファイル状態を表す場合、イベントログから読取られる第 1 の項目は、10 : 01 am に起こったイベントであろう。ステップ 810 において、時刻、イベントのタイプおよびファイル名を含む次のイベントログ項目が読取られる。ステップ 820 において、イベントのタイプがオープンファイルイベントであれば、ステップ 830 において、このファイルを表すように新たなバーが作成され、ファイルが開かれた時に対応して X 軸に対してバーの左側が描かれる。イベントがオープンファイルイベントでなければ、ステップ 840 において、イベントがクローズファイルイベントであることを検証するためにイベントがテストされる。イベントがクローズファイルイベントであれば、ステップ 850 において、クローズイベントから読取られる時点にバーの右側を描くことによってこのファイルのバーが完成され、バーの左側と右側とは接続されて、長方形のバーの上部および底部を形成する。グラフによって表される期間内に起こったセーブファイルイベントが読取られ、セーブが行われた時点と一致する位置におけるバー内に線が引かれ得る。

【 0 0 2 8 】

スナップショット線は、最初は、タイムラインの中央、履歴ファイル内容の調査または履歴環境の作成を要求するためにスナップショット線が用いられた最後の時点、または現在の時刻から過去の特定の時間量を表す時点などの、構成された場所に引かれ得る。

【 0 0 2 9 】

ユーザが例えばスナップショット線を対象の時点にスライドさせることによって対象の時点を選択的に選択すると、ユーザは、点検のためのファイルを示し得る。選択されたファイルの状態の印は、セーブイベントリストにおいて対象の時点を探すことによって検索および表示され得る。セーブイベントリストは、時間によって索引付けされ得る。所望のファイルについての対象の時点に対応するセーブリスト項目が見つけれられてもよく、ファイルの保存された状態を検索するためにファイル参照が用いられ得る。ファイルの保存された状態に基づいて、ファイルの状態が再作成され得る。ある実施例では、セーブイベントリストは、時間によって索引付けされる代わりに、または時間によって索引付けされることに加えて、テキストラベルによって索引付けされてもよい。

【 0 0 3 0 】

図 9 は、本発明の実施例に係る、特定の時点において開いていたファイルの環境を再作成するフロー図である。ステップ 910 において、IDE は、特定の時点においてどのファイルが開いていたかを判断する。特定の時点に対応するイベントログ内の項目の場所が開始場所として用いられる。関連するファイルの群における各ファイルごとに、当該ファイルについて起こった最後のイベントを見つけるために、時間を遡ってログ項目がサーチ

され得る。最後のイベントがオープンイベントであれば、ファイルは開かれ、最後のイベントがクローズイベントであれば、ファイルは閉じられた。ステップ 920 において、各々の開いているファイルごとに、特定の時点より前に、セーブイベントリストにおいて最後のセーブイベント項目が特定される。ステップ 930 において、セーブイベント項目において参照を用いて、ファイル状態データストアからの保存されたファイル状態情報を用いて、ファイル状態が判断され得る。

【0031】

図 10 は、本明細書に記載されているアプローチを実現するために用いられ得るコンピュータシステムの実施例を示す。図 10 に示されるコンピュータシステムは、前に記載したシステムの一部として組込まれてもよく、前に記載した方法を実行するために用いられ得る。例えば、コンピュータシステム 1000 は、ファイル管理、グラフィカルユーザインターフェース、ソフトウェア開発環境を実行するアプリケーションを実行することができる。コンピュータシステム 1000 は、記憶構造 700、710 および 720 などの記憶構造にアクセスし得る。図 10 は、本明細書に記載されているさまざまな実施例によって提供される方法を実行することができるコンピュータシステム 1000 の一実施例の概略図である。なお、図 10 は、単にさまざまな構成要素の一般化された図を提供するよう意図されており、構成要素のうちのいずれかまたは全ては適宜利用可能である。したがって、図 10 は、相対的に分離された態様で、または相対的に統合された態様で個々のシステム要素がいかに実現され得るかを大まかに示している。

【0032】

バス 1005 によって電氣的に結合され得る（または、そうでなければ適宜通信し得る）ハードウェア要素を備えるコンピュータシステム 1000 が示されている。当該ハードウェア要素は、1 つ以上の汎用プロセッサおよび/または 1 つ以上の専用プロセッサ（デジタル信号処理チップ、グラフィックス加速プロセッサなど）を含むがそれらに限定されない 1 つ以上のプロセッサ 1010 と、マウス、キーボードなどを含み得るがそれらに限定されない 1 つ以上の入力装置 1015 と、ディスプレイ装置、プリンタなどを含み得るがそれらに限定されない 1 つ以上の出力装置 1020 とを含み得る。

【0033】

コンピュータシステム 1000 は、1 つ以上の非一時的な記憶装置 1025 をさらに含み得る（および/または 1 つ以上の非一時的な記憶装置 1025 と通信し得る）、当該 1 つ以上の非一時的な記憶装置 1025 は、ローカルおよび/またはネットワークアクセス可能な記憶装置を備え得るがそれに限定されるものではなく、および/または、ディスクドライブ、ドライブアレイ、光学記憶装置、プログラム可能、フラッシュ更新可能などであり得るランダムアクセスメモリ（「RAM」）および/またはリードオンリメモリ（「ROM」）などのソリッドステート記憶装置を含み得るがそれらに限定されるものではない。このような記憶装置は、さまざまなファイルシステム、データベース構造などを含むがそれらに限定されない任意の適切なデータストアを実現するように構成され得る。

【0034】

また、コンピュータシステム 1000 は、通信サブシステム 1030 も含んでもよく、当該通信サブシステム 1030 は、モデム、ネットワークカード（無線または有線）、赤外線通信装置、無線通信装置および/またはチップセット（ブルートゥース（登録商標）装置、802.11 装置、Wi-Fi 装置、Wi-Max 装置、セルラ通信設備など）などを含み得るがそれらに限定されるものではない。通信サブシステム 1030 は、データをネットワーク（一例を挙げると、以下に記載されるネットワークなど）、他のコンピュータシステム、および/または本明細書に記載されているその他の装置と交換することを可能にし得る。多くの実施例では、コンピュータシステム 1000 は、上記の RAM または ROM 装置を含み得るワーキングメモリ 1035 をさらに備えている。

【0035】

また、コンピュータシステム 1000 は、現在のところワーキングメモリ 1035 内に位置するように示されているソフトウェア要素も備えてもよく、当該ソフトウェア要素は

、オペレーティングシステム 1040、デバイスドライバ、実行可能なライブラリ、および/または1つ以上のアプリケーションプログラム 1045などの他のコードを含み、本明細書に記載されているように、さまざまな実施例によって提供されるコンピュータプログラムを備えてもよく、および/または、他の実施例によって提供される方法を実現するようにおよび/またはシステムを構成するように設計され得る。単に一例として、上記の方法に関連して記載される1つ以上の手順は、コンピュータ（および/またはコンピュータ内のプロセッサ）によって実行可能なコードおよび/または命令として実現され得る。そして、ある局面では、このようなコードおよび/または命令は、記載されている方法に従って1つ以上の動作を実行するように汎用コンピュータ（または他の装置）を構成および/または適合させるために用いられ得る。

10

【0036】

一組のこれらの命令および/またはコードは、上記の記憶装置 1025などの非一時的なコンピュータ読取可能な記憶媒体に格納され得る。場合によっては、記憶媒体は、コンピュータシステム 1000などのコンピュータシステム内に組込まれてもよい。他の実施例では、記憶媒体は、コンピュータシステム（例えば、コンパクトディスクなどの取外し可能な媒体）から分離していてもよく、および/または、命令/コードを格納した状態で汎用コンピュータをプログラム、構成および/または適合させるために記憶媒体が使用可能であるようにインストールパッケージに設けられてもよい。これらの命令は、コンピュータシステム 1000によって実行可能な実行可能コードの形態をとってもよく、および/または、（例えば、さまざまな一般に利用可能なコンパイラ、インストールプログラム、圧縮/復元ユーティリティなどのうちのいずれかを用了）コンピュータシステム 1000上でのコンパイルおよび/またはインストール時に実行可能コードの形態をとるソースおよび/またはインストール可能なコードの形態をとってもよい。

20

【0037】

具体的な要求に従って実質的な変形がなされてもよいことは当業者に明らかであろう。例えば、特注のハードウェアも用いられてもよく、および/または、特定の要素がハードウェア、（アプレットなどの高移植性ソフトウェアを含む）ソフトウェアまたはそれら両方で実現されてもよい。さらに、ネットワーク入力/出力装置などの他のコンピューティング装置との接続が利用されてもよい。

【0038】

上記のように、一局面では、いくつかの実施例は、本発明のさまざまな実施例に係る方法を実行するために（コンピュータシステム 1000などの）コンピュータシステムを利用し得る。一組の実施例によれば、このような方法の手順のうちのいくつかまたは全ては、プロセッサ 1010がワーキングメモリ 1035に含まれる（オペレーティングシステム 1040および/またはアプリケーションプログラム 1045などの他のコードに組込まれ得る）1つ以上の命令の1つ以上のシーケンスを実行することによって実行される。このような命令は、記憶装置 1025のうちの1つ以上などの別のコンピュータ読取可能な媒体からワーキングメモリ 1035に読取られ得る。単に一例として、ワーキングメモリ 1035に含まれる命令のシーケンスを実行することにより、本明細書に記載されている方法の1つ以上の手順をプロセッサ 1010に実行させ得る。

30

40

【0039】

本明細書で用いられる「マシン読取可能な媒体」および「コンピュータ読取可能な媒体」という用語は、特定の態様でマシンを動作させるデータを提供することに関わる任意の媒体を指す。コンピュータシステム 1000を用いて実現される実施例では、さまざまなコンピュータ読取可能な媒体は、命令/コードをプロセッサ 1010に提供して実行することに関与し得て、および/または、このような命令/コードを格納および/または搬送するために用いられ得る。多くの実現例では、コンピュータ読取可能な媒体は、物理的な記憶媒体および/または有形の記憶媒体である。このような媒体は、不揮発性媒体または揮発性媒体の形態をとり得る。不揮発性媒体としては、例えば記憶装置 1025などの光

50

学ディスクおよび／または磁気ディスクが挙げられる。揮発性媒体としては、ワーキングメモリ 1035 などのダイナミックメモリが挙げられるがそれに限定されるものではない。

【0040】

物理的なコンピュータ読取可能な媒体および／または実体のあるコンピュータ読取可能な媒体の一般的な形態としては、例えばフロッピー（登録商標）ディスク、フレキシブルディスク、ハードディスク、磁気テープ、またはその他の磁気媒体、CD-ROM、その他の光学媒体、パンチカード、ペーパーテープ、穴のパターンを有するその他の物理媒体、RAM、PROM、EPROM、FLASH-EPROM、その他のメモリチップもしくはカートリッジ、またはコンピュータが命令および／またはコードを読取ることができるその他の媒体が挙げられる。

10

【0041】

コンピュータ読取可能な媒体のさまざまな形態は、1つ以上の命令の1つ以上のシーケンスをプロセッサ 1010 に搬送して実行することに関与し得る。単に一例として、当該命令は、最初は、リモートコンピュータの磁気ディスクおよび／または光学ディスク上で搬送され得る。リモートコンピュータは、当該命令をそのダイナミックメモリにロードして、コンピュータシステム 1000 によって受取られるように、および／または、コンピュータシステム 1000 によって実行されるように、伝送媒体を介して当該命令を信号として送り得る。

【0042】

20

通信サブシステム 1030（および／またはそのコンポーネント）は一般に信号を受取り、次いでバス 1005 が当該信号（および／または信号によって搬送されるデータ、命令など）をワーキングメモリ 1035 に搬送し得て、プロセッサ 1010 がワーキングメモリ 1035 から当該命令を検索および実行する。ワーキングメモリ 1035 によって受取られた命令は、プロセッサ 1010 による実行の前または後に、非一時的な記憶装置 1025 に任意に格納され得る。

【0043】

上記の方法、システムおよび装置は一例である。さまざまな構成では、さまざまな手順または構成要素を適宜省略、置換または追加することができる。例えば、代替的な構成では、当該方法は、記載されている順序とは異なる順序で実行されてもよく、および／または、さまざまな段階が追加、省略および／または組合わせられてもよい。また、特定の構成に関連して記載されている特徴は、さまざまな他の構成では組合わせられてもよい。当該構成のさまざまな局面および要素は、同様に組合わせられてもよい。また、技術は進化するので、要素のうちの多くは一例であり、本開示の範囲または特許請求の範囲を限定するものではない。

30

【0044】

（実現例を含む）例示的な構成を十分に理解してもらうために、本明細書には具体的な詳細が記載されている。しかし、これらの具体的な詳細がなくても構成を実施することができる。例えば、構成を曖昧にすることを回避するために、不必要な詳細なしに、周知の回路、プロセッサ、アルゴリズム、構造および技術が示されてきた。本明細書は、例示的な構成を提供しているに過ぎず、請求項の範囲、適用可能性または構成を限定するものではない。それどころか、構成の上記の説明は、記載されている技術を実現するための実施可能に十分な記載を当業者に提供する。本開示の精神または範囲から逸脱することなく、要素の機能および配置の点でさまざまな変更がなされてもよい。

40

【0045】

また、構成は、フロー図またはブロック図として示されるプロセスとして記載されてもよい。各々は動作を逐次的なプロセスとして説明し得るが、動作のうちの多くは並列または同時に実行されてもよい。また、動作の順序は再編成されてもよい。プロセスは、図に含まれていない追加のステップを有していてもよい。さらに、当該方法の例は、ハードウェア、ソフトウェア、ファームウェア、ミドルウェア、マイクロコード、ハードウェア記

50

述言語またはそれらの任意の組合わせによって実現されてもよい。ソフトウェア、ファームウェア、ミドルウェアまたはマイクロコードで実現される場合、必要なタスクを実行するためのプログラムコードまたはコードセグメントは、記憶媒体などの非一時的なコンピュータ読取可能な媒体に格納され得る。プロセッサが、記載されているタスクを実行し得る。

【 0 0 4 6 】

一実施例によれば、装置であって、一組の関連するファイルのグラフをコンピュータディスプレイ上に表示するための手段を備え、上記グラフは、上記一組の関連するファイルの各ファイルごとに、開始ポイントおよび終了ポイントおよび1つ以上の中間チェックポイントを示し、上記開始ポイント、上記終了ポイントおよび上記1つ以上の中間チェックポイントは各々、ある時点を表し、各時点は、上記各ファイルのファイル内容の状態に関連付けられ、上記各ファイルは、第1の時点において現在の状態の内容を有し、上記装置はさらに、上記第1の時点より前の特定の時点を示す入力を受取るための手段と、上記一組の関連するファイルのファイル内容を備える環境を再生するための手段とを備え、上記環境における上記ファイル内容は、上記特定の時点に関連付けられた履歴状態を有する、装置が提供される。

10

【 0 0 4 7 】

一例では、上記装置は、上記環境を表示するための手段をさらに備え、上記環境を表示することは、上記特定の時点において開いていたファイル内容の状態の印を表示することを含む。

20

【 0 0 4 8 】

一例では、上記装置は、上記各ファイルの上記開始ポイント、上記終了ポイントおよび上記1つ以上の中間チェックポイントに対応する時点を含む履歴上の時点を含むタイムラインを表示するための手段をさらに備える。

【 0 0 4 9 】

一例では、上記一組の関連するファイルの上記グラフは、上記一組の関連するファイルにおける上記各ファイルを表す1つ以上のバーを有する棒グラフであり、上記1つ以上のバーのうちの1つのバーの一方の側は、上記各ファイルの上記開始ポイントに対応する上記タイムライン上のある時点に対して位置決めされ、上記バーのもう一方の側は、上記各ファイルの上記終了ポイントに対応する上記タイムライン上のある時点に対して位置決めされ、上記バー内に1つ以上の印を表示することを備え、各印は、上記各ファイルの中間チェックポイントに対応する上記タイムライン上のある時点に対して位置決めされる。

30

【 0 0 5 0 】

一例では、上記タイムラインは、水平な線として表示され、各々がファイルを表す上記1つ以上のバーは、履歴上のある時点において上記タイムラインと交差する垂直な線が、上記履歴上のある時点に対応する側における各バーまたは印とも交差するように、上記タイムラインに平行に表示される。

【 0 0 5 1 】

一例では、上記タイムライン上の特定の時点を示す入力を受取るための手段は、上記タイムラインに直交する線を表示するための手段をさらに備え、上記直交する線は、履歴スナップショットを時間で表し、上記タイムライン上の特定の時点を示す入力を受取るための手段はさらに、入力の受取りに応答して、上記直交する線が上記特定の時点において上記タイムラインと交差するように移動することを表示するための手段を備える。

40

【 0 0 5 2 】

一例では、ファイルの上記開始ポイントは、上記一組の関連するファイルにおける上記各ファイルが開かれる時刻を表し、上記ファイルの上記終了ポイントは、上記各ファイルが閉じられる時刻を表し、上記各ファイルの上記中間チェックポイントの各中間チェックポイントは、上記各ファイルが保存された時刻を表す。

【 0 0 5 3 】

一例では、上記グラフを表示するための手段は、特定の期間中に開いていたファイルの

50

第1のリストを検索するための手段と、上記特定の期間内の任意の時刻において開いていた上記一組の関連するファイルにおける各ファイルごとに、第1のファイルオープン時点として、上記各ファイルが開かれた上記特定の期間内の時刻を検索するための手段と、第1のファイルクローズ時点として、上記各ファイルが閉じられた対応する時刻を検索するための手段と、上記第1のファイルオープン時点に従って一方の側が位置決めされ、上記第1のファイルクローズ時点に従ってもう一方の側が位置決めされた第1のバーを表示するための手段と、第1のファイルセーブ時点として、上記各ファイルが保存された1つ以上の時刻を検索するための手段とをさらに備え、第1のファイルセーブ時点は、上記第1のファイルオープン時点と上記第1のファイルクローズ時点との間にあり、上記グラフを表示するための手段はさらに、上記第1のバー上に1つ以上の印を表示するための手段を備え、各印は、上記1つ以上の第1のファイルセーブ時点のうちの1つに対応する。

10

【0054】

一例では、上記グラフを表示するための手段は、上記ファイルの第1のリストにおける各ファイルごとに、第2のファイルオープン時点として、上記各ファイルが開かれた上記特定の期間内の時刻を検索するための手段と、第2のファイルクローズ時点として、上記各ファイルが閉じられた対応する時刻を検索するための手段とをさらに備え、上記第2のファイルオープン時点は、上記第1のファイルクローズ時点の後であり、上記グラフを表示するための手段はさらに、上記第2のファイルオープン時点に従って一方の側が位置決めされ、上記第2のファイルクローズ時点に従ってもう一方の側が位置決めされた第2のバーを表示するための手段と、第2のファイルセーブ時点として、上記各ファイルが保存された1つ以上の時刻を検索するための手段とをさらに備え、第2のファイルセーブ時点は、上記第2のファイルオープン時点と上記第2のファイルクローズ時点との間にあり、上記グラフを表示するための手段はさらに、上記第2のバー上に1つ以上の印を表示するための手段を備え、各印は、上記1つ以上の第2のファイルセーブ時点のうちの1つに対応する。

20

【0055】

一例では、上記ファイルの第1のリストを検索するための手段は、時間ベースの索引に基づく。

【0056】

一例では、上記第1のファイルセーブ時点を検索するための手段は、時間ベースの索引以外の表から上記第1のファイルセーブ時点を検索することをさらに備える。

30

【0057】

一例では、上記一組の関連するファイルは、共通のプロジェクトに属しており、上記一組の関連するファイルにおける複数の上記ファイルは、共通のディレクトリ内にあり、上記一組の関連するファイルの少なくとも1つのファイルは、上記共通のディレクトリの外側にある。

【0058】

一例では、上記一組の関連するファイルは、統合開発環境内で管理されるソースファイルを備え、各ソースファイルは、コンピュータプログラミング言語に準拠するテキストを含む。

40

【0059】

一例では、上記装置は、上記統合開発環境をさらに備え、上記統合開発環境において1つ以上の開いているファイルを保存するためにファイルセーブ時点においてトリガを受取るための手段と、上記グラフにおいて上記ファイルセーブ時点を特定するためのラベルを受取るための手段と、上記保存された1つ以上の開いているファイルを上記ラベルに基づいて索引付けするために索引項目を作成するための手段とをさらに備える。

【0060】

一例では、上記ラベルは、上記トリガが受取られた時を示すタイムスタンプである。

一例では、上記ラベルは、タイムスタンプではないマイルストンの名前である。

【0061】

50

一例では、上記トリガは、上記１つ以上の開いているファイルの保存を求めるユーザ要求に応答して生成される。

【００６２】

一例では、上記トリガは、タイマに基づいて生成される。

いくつかの例示的な構成について説明してきたが、本開示の精神から逸脱することなく、さまざまな修正例、代替的な構造および等価物が用いられてもよい。例えば、上記の要素はより大きなシステムの構成要素であってもよく、そのような場合には他の規則が本発明の適用例に優先してもよく、またはそうでなければ本発明の適用例を修正してもよい。また、上記の要素が検討される前、最中または後にいくつかのステップが着手されてもよい。したがって、上記の説明は特許請求の範囲を制限するものではない。

10

【図１】

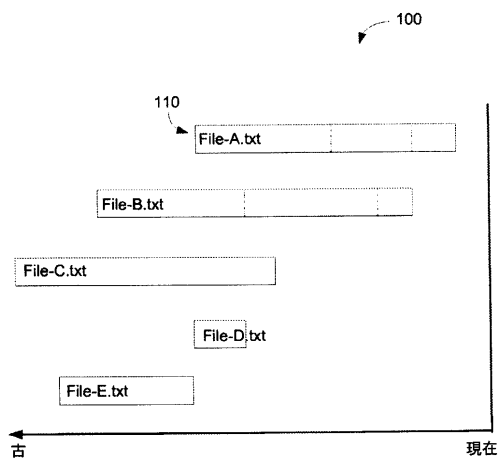


FIG. 1

【図２】

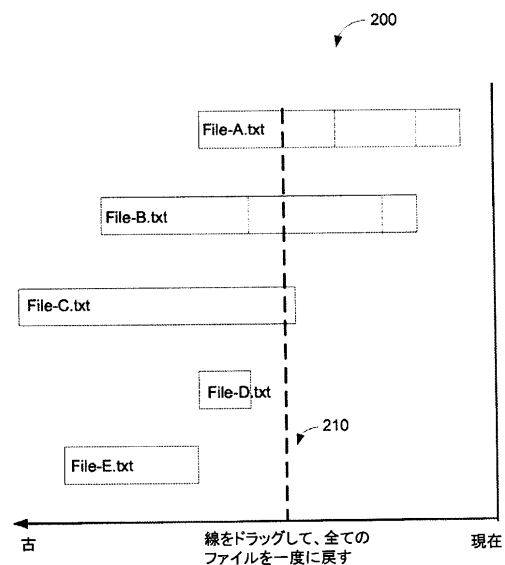


FIG. 2

【図 3 a】

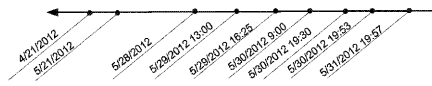


FIG. 3a

【図 3 b】

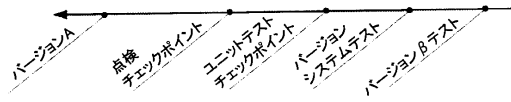


FIG. 3b

【図 3 c】



FIG. 3c

【図 4】

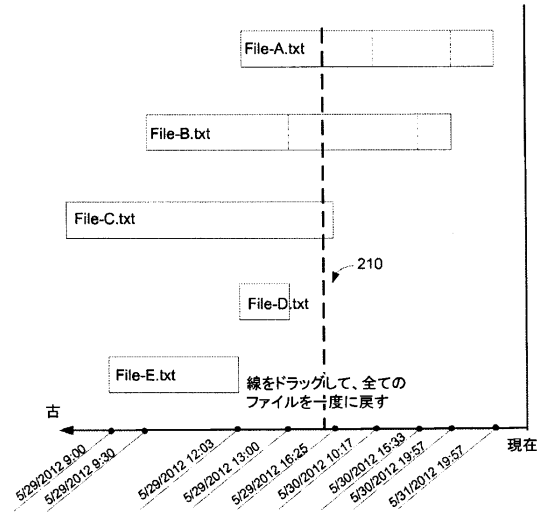


FIG. 4

【図 5】

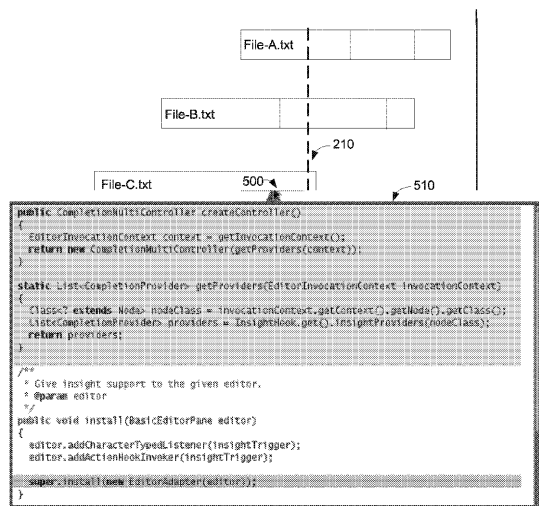


FIG. 5

【図 6】

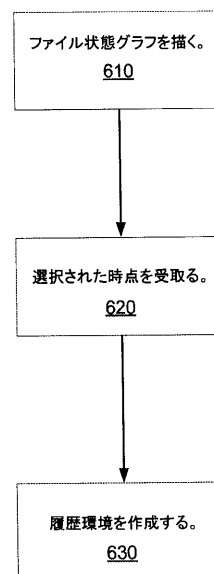


FIG. 6

【図 7】

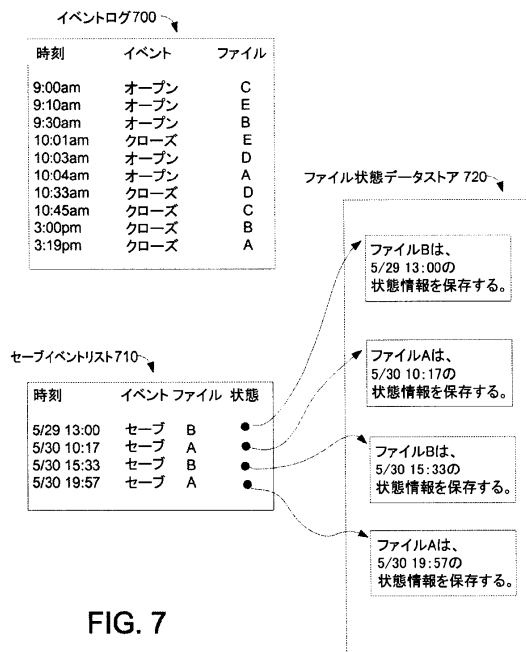


FIG. 7

【図 8】

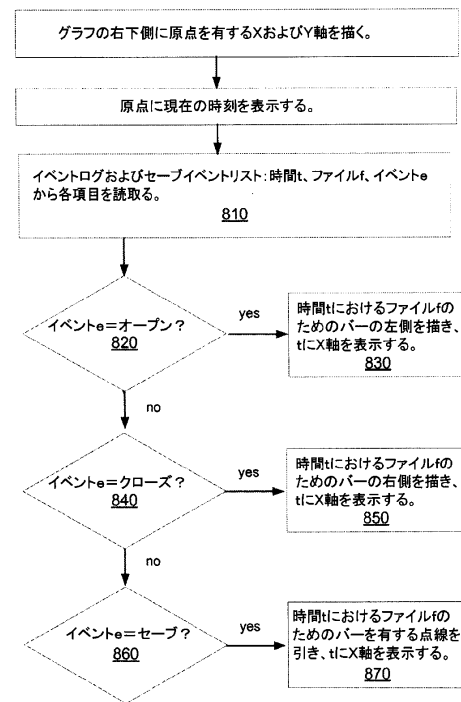


FIG. 8

【図 9】

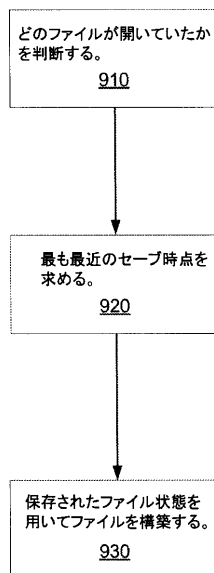


FIG. 9

【図 10】

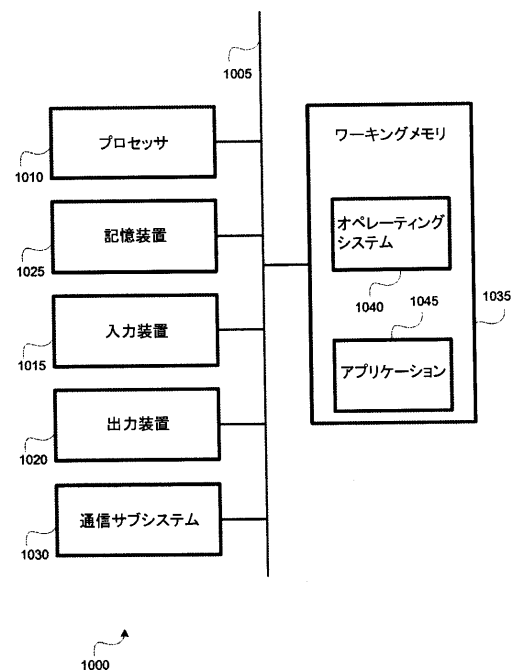


FIG. 10

フロントページの続き

- (56)参考文献 特開2007-299289(JP,A)
米国特許出願公開第2006/0265249(US,A1)
米国特許出願公開第2004/0153456(US,A1)
特表2009-506395(JP,A)
米国特許出願公開第2012/0144335(US,A1)
大森 隆行, " プログラム変更履歴調査のための編集操作再生器 ", レクチャーノート/ソフトウェア学36 ソフトウェア工学の基礎XVII, 日本, 株式会社近代科学社, 2010年11月30日, p. 45 - 54

- (58)調査した分野(Int.Cl., DB名)
G06F 12/00