

(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2014-41403

(P2014-41403A)

(43) 公開日 平成26年3月6日(2014.3.6)

(51) Int. Cl. F I テーマコード (参考)  
**G06F 9/44 (2006.01)** G06F 9/06 620J 5B376  
**G06Q 10/06 (2012.01)** G06F 17/60 162A

審査請求 未請求 請求項の数 8 O L (全 27 頁)

(21) 出願番号 特願2012-182071 (P2012-182071)  
 (22) 出願日 平成24年8月21日 (2012.8.21)

(71) 出願人 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 100092978  
 弁理士 真田 有  
 (74) 代理人 100112678  
 弁理士 山本 雅久  
 (72) 発明者 車 義孝  
 神奈川県川崎市中原区上小田中4丁目1番1号 富士通アドバンステクノロジー株式会社内  
 (72) 発明者 鈴木 智久  
 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

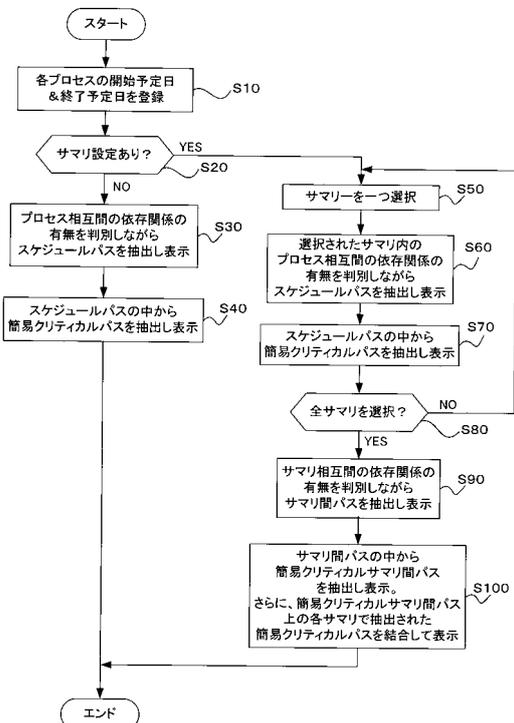
(54) 【発明の名称】 スケジュール管理プログラム、スケジュール管理装置、及びスケジュール管理方法

(57) 【要約】

【課題】プロセスの前後関係の精度を大きく崩すことなくスケジュールパスを簡易的に作成する。

【解決手段】複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミングを予め格納部に登録し、格納部に登録された開始予定タイミングおよび終了予定タイミングに基づき、複数のプロセス相互間の依存関係の有無を判別し、前記複数のプロセスのうち、依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する。

【選択図】 図6



**【特許請求の範囲】****【請求項 1】**

複数のプロセスを含む作業のスケジュールを管理するコンピュータに、  
前記複数のプロセスのそれぞれについて予め登録された開始予定タイミングおよび終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、  
前記複数のプロセスのうち、前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、  
処理を実行させる、スケジュール管理プログラム。

**【請求項 2】**

前記複数のプロセスのうちの判別対象の第 1 プロセスと開始予定タイミングが当該第 1 プロセスの終了予定タイミング以降である第 2 プロセスとの間に前記依存関係が有ると判別する、  
処理を前記コンピュータに実行させる、請求項 1 記載のスケジュール管理プログラム。

10

**【請求項 3】**

抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する、  
処理を前記コンピュータに実行させる、請求項 1 または請求項 2 に記載のスケジュール管理プログラム。

**【請求項 4】**

接続されたプロセスの段数が最も多いスケジュールパスを前記簡易クリティカルパスとして抽出する、  
処理を前記コンピュータに実行させる、請求項 3 記載のスケジュール管理プログラム。

20

**【請求項 5】**

前記接続されたプロセスの段数が最も多いスケジュールパスが複数本抽出された場合、当該複数本の同一段数のスケジュールパスのうち、期間の最も長いスケジュールパスを前記簡易クリティカルパスとして抽出する、  
処理を前記コンピュータに実行させる、請求項 4 記載のスケジュール管理プログラム。

**【請求項 6】**

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出し、  
前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、算出された前記差分だけシフトする、  
処理を前記コンピュータに実行させる、請求項 1 ~ 請求項 7 のいずれか一項に記載のスケジュール管理プログラム。

30

**【請求項 7】**

複数のプロセスを含む作業のスケジュールを管理するスケジュール管理装置であって、  
前記複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミングを予め登録される格納部と、  
前記格納部に登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別する第 1 判別部と、  
前記複数のプロセスのうち、前記第 1 判別部によって前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する第 1 抽出部と、を有する、スケジュール管理装置。

40

**【請求項 8】**

複数のプロセスを含む作業のスケジュールを管理するスケジュール管理方法であって、  
前記複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミング

50

を予め格納部に登録し、

前記格納部に登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、

前記複数のプロセスのうち、前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、スケジュール管理方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、スケジュール管理プログラム、スケジュール管理装置、及びスケジュール管理方法に関する。

10

【背景技術】

【0002】

製品開発などの設計作業は、細分化された設計プロセスのフローによって実現され運用されている。細分化された各設計プロセス（以下、単にプロセスという）は、他のプロセスとの依存関係を有しており、当該依存関係に従って順序だてられて実施されることにより、上記設計作業が進められる。

【0003】

プロセスの相互間には、様々な依存関係（要素）が有る。その要素としては、例えば、

- (1)各プロセスで得られる成果物相互間の入出力関係
- (2)プロセス相互間の依存関係
- (3)スケジュール（工数）
- (4)担当者の割当て状況
- (5)作業の優先順位／リスク

20

等が挙げられる。

【0004】

オペレータ等は、プロセス毎に上述のような多数の要素をデータベース等のデータ格納部に入力登録し、登録した要素に基づき、手作業、または、設計スケジュールの工程を管理するプロジェクト管理支援ツール等によってクリティカルパスを抽出生成する。クリティカルパスは、設計作業のスケジュールに影響を与える可能性の高い、管理上、重要なパス（複数のプロセスからなるフロー）である。抽出されたクリティカルパスは、ディスプレイ（クリティカルパス表示部）上で、ガントチャートとして、プロセスの実行スケジュールとともに表示される。

30

【0005】

プロジェクトのマネージャ等は、上述のように表示されたクリティカルパス（プロジェクト内における重要パス）をディスプレイの画面上で参照し、設計作業のスケジュールの管理に活用している。

【先行技術文献】

【特許文献】

【0006】

【特許文献1】特開平08-30679号公報

40

【特許文献2】特開平11-224238号公報

【発明の概要】

【発明が解決しようとする課題】

【0007】

ところで、オペレータ等によって行なわれる、各プロセスの依存関係情報（例えば上記要素(1)~(5)）を入力登録する作業は、1プロジェクト当たり100~200プロセスについて実行しなければならず、その作業には、1日から長いものでは1週間以上を要する場合がある。

【0008】

特に、多段のプロセスからなる作業の工程管理（スケジュール管理）を行なうには、オ

50

ペレータ等は、事前に、プロセス相互間の依存関係（或るプロセスが完了しなければ、別の或るプロセスを開始しないといった関係）を確認した上で、プロセスのパス（スケジュールパス）を作成しなければならない。この依存関係の確認には膨大な工数や時間が必要である。

【0009】

成果物相互間の入出力関係や、プロセス相互間の依存関係については、その関係が複雑なため、入力登録作業時に入力ミスが発生しやすい。

また、入力登録作業は、スケジュール変更の度に発生し、円滑なプロジェクト運用の妨げにもなっている。

【0010】

一つの側面で、本発明は、プロセスの前後関係の精度を大きく崩すことなくプロセスのスケジュールパスを簡易的に作成することを目的とする。

【課題を解決するための手段】

【0011】

一つの案において、スケジュール管理プログラムは、複数のプロセスを含む作業のスケジュールを管理するコンピュータに、前記複数のプロセスのそれぞれについて予め登録された開始予定タイミングおよび終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、前記複数のプロセスのうち、前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、処理を実行させる。

【発明の効果】

【0012】

一実施形態によれば、プロセスの前後関係の精度を大きく崩すことなくプロセスのスケジュールパスを簡易的に作成することができる。

【図面の簡単な説明】

【0013】

【図1】第1実施形態のスケジュール管理装置のハードウェア構成および機能構成を示すブロック図である。

【図2】開始・完了日設定画面の例を示す図である。

【図3】図1に示す格納部に登録保存されるプロセス管理リストの例を示す図である。

【図4】サマ리를説明する図である。

【図5】図1に示すスケジュール管理装置によって抽出表示されるスケジュールパスおよび簡易クリティカルパスの例を示す図である。

【図6】図1に示すスケジュール管理装置の動作を説明するフローチャートである。

【図7】図1に示すスケジュール管理装置によるスケジュールパス（サマリ間パス）の抽出動作を説明するフローチャートである。

【図8】図1に示すスケジュール管理装置による簡易クリティカルパス（簡易クリティカルサマリ間パス）の抽出動作を説明するフローチャートである。

【図9】図1に示すスケジュール管理装置の動作を具体的に説明する図である。

【図10】図1に示すスケジュール管理装置の動作を具体的に説明する図である。

【図11】図1に示すスケジュール管理装置の動作を具体的に説明する図である。

【図12】図1に示すスケジュール管理装置の動作を具体的に説明する図である。

【図13】図1に示すスケジュール管理装置の動作を具体的に説明する図である。

【図14】第2実施形態のスケジュール管理装置のハードウェア構成および機能構成を示すブロック図である。

【図15】図14に示すスケジュール管理装置で用いられるプロセス管理リストの例を示す図である。

【図16】図14に示すスケジュール管理装置の動作を説明するフローチャートである。

【図17】図14に示すスケジュール管理装置の動作を具体的に説明する図である。

【図18】図14に示すスケジュール管理装置の動作を具体的に説明する図である。

【図19】図14に示すスケジュール管理装置の動作を具体的に説明する図である。

10

20

30

40

50

【図20】(A)，(B)は図14に示すスケジュール管理装置の動作を具体的に説明する図である。

【図21】(A)，(B)は図14に示すスケジュール管理装置の動作を具体的に説明する図である。

【図22】図14に示すスケジュール管理装置の動作を具体的に説明する図である。

【発明を実施するための形態】

【0014】

以下、図面を参照して実施の形態を説明する。

〔1〕第1実施形態

〔1-1〕第1実施形態のスケジュール管理装置の構成および機能

10

図1は、第1実施形態のスケジュール管理装置1のハードウェア構成および機能構成を示すブロック図である。

【0015】

図1に示すスケジュール管理装置1は、複数のプロセスを含む作業のスケジュールを管理するもので、例えば、設計スケジュールの工程を管理するプロジェクト管理支援ツールである。また、スケジュール管理装置1は、一般的なパーソナルコンピュータ等によって構成され、バス50経由で相互に通信可能に接続された登録部10、格納部20、処理部30および表示部40を有している。

【0016】

登録部10は、オペレータ等によって操作されるキーボード、マウス等を含むマンマシンインタフェースであり、各プロセスの開始予定日(開始予定タイミング)および終了予定日(終了予定タイミング)を入力登録する。その際、オペレータ等は、表示部40に表示された、図2に示すような開始・完了日設定画面(ポップアップウィンドウ)を参照し、当該画面におけるプロセス毎の開始予定日/終了予定日の欄に、キーボード、マウス等によって開始予定日/終了予定日を記入する。オペレータ等は、全てのプロセス(図2に示す例ではプロセス1~プロセスn(nは4以上の自然数))の開始予定日/終了予定日を入力すると、開始・完了日設定画面における登録ボタンをクリックする。これに伴い、プロセス毎の欄に登録された開始予定日/終了予定日が、格納部20に保存されるプロセス管理リストに登録される。なお、図2は開始・完了日設定画面(ポップアップウィンドウ)の例を示す図である。

20

30

【0017】

プロセス管理リストには、例えば図3に示すように、ID、工程名、開始日、終了日、日数、サマリID、依存工程IDが、要素として登録されている。図3は、図1に示す格納部20に登録保存されるプロセス管理リストの例を示す図である。

【0018】

IDは、各プロセスに割り当てられる識別番号(自動連番;1,2,...,n)である。工程名は、各プロセスの名称(タスク名;プロセス1~プロセスn)である。開始日は、各プロセスの開始予定日(年月日)であり、終了日は、各プロセスの終了予定日(年月日)である。日数は、開始日と終了日とに基づき自動算出される期間である。なお、本実施形態では、開始予定タイミング/終了予定タイミングを開始予定日/終了予定日(年月日)として登録し、各プロセスの期間を日数として登録しているが、開始予定タイミング/終了予定タイミングを開始予定日時/終了予定日時として登録し、各プロセスの期間を時間単位または分単位として登録してもよい。

40

【0019】

サマリIDは、各プロセスの上位に各プロセスを管理するサマリがある場合に各プロセスが属するサマリを特定する識別番号である。ここで、サマリとは、図4に示すように、プロジェクト内における各プロセスをグループ化したものである。図4はサマリを説明する図で、図4では、サマリ1に3つのプロセスA~Cが属し、サマリ2に2つのプロセスD,Eが属している例が示されている。また、図3では、サマリIDがSM1であるサマリにプロセス1,2が属し、サマリIDがSM2であるサマリにプロセスnが属している

50

例が示されている。各プロセスの上位に各プロセスを管理するサマリがある場合、サマリIDを設定することにより各プロセスはサマリにグループ分けされる一方、各プロセスの上位に各プロセスを管理するサマリがない場合、サマリIDは設定されない。

#### 【0020】

依存工程IDは、プロセス相互間の依存関係が事前に決められている場合に依存先プロセスを特定する識別番号である。図3では、プロセス1がプロセス2に依存する例が示されている。プロセス相互間の依存関係が事前に決められていない場合、依存工程IDは設定されない。

なお、図3では図示していないが、プロセス管理リストには、各プロセスの担当者名も要素として登録される。

#### 【0021】

本実施形態において、上述したプロセス管理リストに登録される要素のうち、開始日（開始予定日）および終了日（終了予定日）の2つだけが、オペレータ等によって登録部10から入力される。日数については、上述した通り、開始日（開始予定日）および終了日（終了予定日）に基づき自動算出される。また、ID、工程名、サマリID、依存工程ID、担当者名は、工程作成時に事前に定義される。

#### 【0022】

格納部20は、RAM（Random Access Memory）、HDD（Hard Disk Drive）、SSD（Solid State Drive）等の内部記憶装置であってもよいし、外部記憶装置であってもよい。格納部20は、少なくとも、図3を参照しながら上述したプロセス管理リストや、コンピュータ（処理部30）をスケジュール管理装置1として動作させるスケジュール管理プログラムとを記憶する。

#### 【0023】

処理部（CPU（Central Processing Unit）、コンピュータ、プロセッサ）30は、格納部20に記憶されたスケジュール管理プログラムを読み出して実行することで、第1判別部31、第1抽出部32、第2抽出部33、第2判別部34、第3抽出部35、第4抽出部36および表示制御部37としての機能を果たす。

#### 【0024】

第1判別部31、第1抽出部32および第2抽出部33は、プロセスがサマリにグループ化されている場合のサマリ内のプロセス、もしくは、プロセスがサマリにグループ化されていない場合のプロセスについて、スケジュールパス（後述）や簡易クリティカルパス（後述）を生成抽出するサマリ内パス生成部30aを構成している。

#### 【0025】

第2判別部34、第3抽出部35および第4抽出部36は、プロセスがサマリにグループ化されている場合のサマリについて、サマリ間パス（後述）や簡易クリティカルサマリ間パス（後述）を生成抽出するサマリ間パス生成部30bを構成している。なお、第2判別部34、第3抽出部35および第4抽出部36は、後述するように、第1判別部31、第1抽出部32および第2抽出部33が各プロセスに対して行なっている処理と同様の処理を、各サマリに対して行なっている。つまり、サマリ間パス生成部30bにおけるサマリ、サマリ間パスおよび簡易クリティカルサマリ間パスは、それぞれ、サマリ内パス生成部30aにおけるプロセス、スケジュールパスおよび簡易クリティカルパスに相当する。

#### 【0026】

第1判別部31は、格納部20（プロセス管理リスト）に登録された開始予定日および終了予定日に基づき、複数のプロセス相互間の依存関係の有無を判別する。特に、第1判別部31は、複数のプロセスのうちの判別対象の第1プロセスと、開始予定日が当該第1プロセスの終了予定日以降である第2プロセスとの間に依存関係が有ると判別する。

#### 【0027】

第1抽出部32は、複数のプロセスのうち、第1判別部31によって依存関係が有ると判別されたプロセスどうしを数珠繋ぎに接続し複数のスケジュールパスを抽出する。表示部40において、依存関係の有る第1プロセスと第2プロセスとの間は、図5に示すよう

10

20

30

40

50

に、依存線（細線矢印参照）によって接続され、依存線によって接続されたプロセスがスケジュールパスとして抽出され表示される。なお、第1判別部31および第1抽出部32によるスケジュールパスの抽出動作については、図7および図9～図12を参照しながら詳述する。

#### 【0028】

第2抽出部33は、第1抽出部32によって抽出された複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する。特に、第2抽出部33は、接続されたプロセスの段数が最も多いスケジュールパスを簡易クリティカルパスとして抽出する。さらに、第2抽出部33は、接続されたプロセスの段数が最も多いスケジュールパスが複数本抽出された場合、当該複数本の同一段数のスケジュールパスのうち、期間（日数）の最も長いスケジュールパスを簡易クリティカルパスとして抽出する。つまり、第2抽出部33は、プロセスの接続段数が最も多いパス（パス内の構成が複雑なパス）を重要と判断し、さらに、プロセスの接続段数が最も多い同一段数のパスが複数本抽出された場合、期間の長いものを重要と判断している。

10

#### 【0029】

なお、プロセスの段数も期間（日数）も同じ複数本のスケジュールパスが抽出された場合、第2抽出部33は、当該複数本のスケジュールパスを全て簡易クリティカルパスとして抽出する。

また、図4に示すごとく複数のプロセスが複数のサマリにグループ化されている場合、第2抽出部33は、第1判別部31および第1抽出部32と協働して、各サマリに含まれる2以上のプロセスについて、上述と同様にしてスケジュールパスおよび簡易クリティカルパスを抽出する。なお、第2抽出部33による簡易クリティカルパスの抽出動作については、図8および図13を参照しながら後述する。

20

#### 【0030】

第2判別部34は、複数のサマリのそれぞれについて、各サマリに含まれる2以上のプロセスについて格納部20に予め登録された開始予定日（2以上のプロセスの開始予定日のうち最も早いもの）および終了予定日（2以上のプロセスの終了予定日のうち最も遅いもの）に基づき、複数のサマリ相互間の依存関係の有無を判別する。特に、第2判別部34は、複数のサマリのうちの判別対象の第1サマリと、最も早い開始予定日が当該第1サマリの最も遅い終了予定日以降である第2サマリとの間に依存関係があると判別する。

30

#### 【0031】

第3抽出部35は、複数のサマリのうち、第2判別部34によって依存関係があると判別されたサマリどうしを接続して複数のサマリ間パス（プロセス間のスケジュールパスに相当）を抽出する。表示部40において、依存関係の有る第1サマリと第2サマリとの間は、図4に示すように、依存線（細線矢印参照）によって接続され、依存線によって接続されたサマリがサマリ間として抽出され表示される。なお、第2判別部34および第3抽出部35によるサマリ間パスの抽出動作については、図7および図9～図12を参照しながら詳述する。

#### 【0032】

第4抽出部36は、第3抽出部35によって抽出された複数のサマリ間パスのうち、サマリ構成の複雑なサマリ間パスを簡易クリティカルサマリ間パスとして抽出する。特に、第4抽出部36は、接続されたサマリの段数が最も多いサマリ間パスを簡易クリティカルサマリ間パスとして抽出する。さらに、第4抽出部36は、接続されたサマリの段数が最も多いサマリ間パスが複数本抽出された場合、当該複数本の同一段数のサマリ間パスのうち、期間（日数）の最も長いサマリ間パスを簡易クリティカルサマリ間パスとして抽出する。なお、第4抽出部36による簡易クリティカルサマリ間パスの抽出動作については、図8および図13を参照しながら後述する。

40

#### 【0033】

表示制御部37は、表示部40の表示状態を制御する。特に、表示制御部37は、図2に示すような開始・完了日設定画面や、第1抽出部32によって抽出されたスケジュール

50

パスや、第2抽出部33によって抽出された簡易クリティカルパスや、第3抽出部35によって抽出されたサマリ間パスや、第4抽出部36によって抽出された簡易クリティカルサマリ間パスを、図5に示すごとく、表示部40において表示する。

#### 【0034】

なお、図5は、図1に示すスケジュール管理装置1によって抽出表示されるスケジュールパスおよび簡易クリティカルパスの例を示す図である。図5に示す表示例では、画面左側に、格納部20に登録されたプロセス管理リストの情報が表示されるとともに、画面右側に、第1抽出部32によって抽出されたスケジュールパスと、第2抽出部33によって抽出された簡易クリティカルパスとがガントチャートによって表示されている。そして、図5のガントチャートでは、2本のスケジュールパス、つまり、プロセス1, 2, 6を依存線（細線矢印参照）で接続したパスと、プロセス4, 5, 3を依存線（細線矢印参照）で接続したパスとが表示されている。さらに、これら2本のスケジュールパスのうち、プロセス4, 5, 3を依存線で接続したパスが簡易クリティカルパス（太線矢印参照）として表示されている。

10

#### 【0035】

〔1-2〕第1実施形態のスケジュール管理装置の動作

次に、図6～図13を参照しながら、上述のごとく構成された第1実施形態のスケジュール管理装置1の動作について説明する。

まず、図6に示すフローチャート（ステップS10～S100）に従って、図1に示すスケジュール管理装置1の動作について説明する。

20

#### 【0036】

スケジュールを開始するにあたり、オペレータ等は、表示部40に表示された開始・完了日設定画面（図2）を参照しながら、登録部10に含まれるキーボード、マウス等を操作し、各プロセスの開始予定日および終了予定日を入力する。これにより、格納部20に保存されるプロセス管理リストに、例えば図3に示すように、各プロセスの開始予定日および終了予定日が移入され登録される（ステップS10）。

#### 【0037】

そして、処理部30は、サマリ設定が行なわれているか否か、つまりプロセス管理リストにおいてサマリIDが設定されているか否かを判定する（ステップS20）。

サマリが設定されていなければ（ステップS20のNORルート）、以下の処理（ステップS30, S40）が実行される。

30

#### 【0038】

つまり、第1判別部31が、格納部20に登録された開始予定日および終了予定日に基づき複数のプロセス相互間の依存関係の有無を判別しながら、第1抽出部32が、第1判別部31によって依存関係が有ると判別されたプロセスどうしを数珠繋ぎに接続し複数のスケジュールパスを抽出する。抽出されたスケジュールパスは、表示制御部37により表示部40に表示される（ステップS30）。このステップS30の処理の詳細については、図7および図9～図12を参照しながら後述する。

#### 【0039】

この後、第2抽出部33は、第1抽出部32によって抽出された複数のスケジュールパスから、簡易クリティカルパスを抽出し、抽出された簡易クリティカルパスは、表示制御部37により表示部40に表示され（ステップS40）、処理部30は処理を終了する。

40

一方、サマリが設定されてる場合（ステップS20のYESルート）、以下の処理（ステップS50～S100）が実行される。

#### 【0040】

つまり、処理部30は、複数のサマリの中から一つのサマリを選択する（ステップS50）。選択されたサマリに含まれる2以上のプロセスについて、第1判別部31が、格納部20に登録された開始予定日および終了予定日に基づき複数のプロセス相互間の依存関係の有無を判別しながら、第1抽出部32が、第1判別部31によって依存関係が有ると判別されたプロセスどうしを数珠繋ぎに接続し複数のスケジュールパスを抽出する。抽出

50

されたスケジュールパスは、表示制御部 37 により表示部 40 に表示される（ステップ S 60）。このステップ S 60 の処理の詳細については、図 7 および図 9 ~ 図 12 を参照しながら後述する。

【0041】

この後、第 2 抽出部 33 は、第 1 抽出部 32 によって抽出された複数のスケジュールパスから、簡易クリティカルパスを抽出し、抽出された簡易クリティカルパスは、表示制御部 37 により表示部 40 に表示される（ステップ S 70）。

そして、処理部 30 は、全てのサマリが選択されたか否か、つまり全てのサマリに対しステップ S 50 ~ S 70 の処理が実行されたか否かを判定する（ステップ S 80）。全てのサマリが選択されていない場合（ステップ S 80 の NO ルート）、処理部 30 は、ステップ S 50 の処理に戻る。一方、全てのサマリが選択されている場合（ステップ S 80 の YES ルート）、処理部 30 は、ステップ S 90 の処理に移行する。

【0042】

ステップ S 90 では、第 2 判別部 34 が、各サマリにおける最も早い開始予定日および最も遅い終了予定日に基づき複数のサマリ相互間の依存関係の有無を判別しながら、第 3 抽出部 35 が、第 2 判別部 34 によって依存関係が有ると判別されたサマリどうしを数珠繋ぎに接続し複数のサマリ間パスを抽出する。抽出されたサマリ間パスは、表示制御部 37 により表示部 40 に表示される。このステップ S 90 の処理の詳細については、図 7 および図 9 ~ 図 12 を参照しながら後述する。

【0043】

この後、第 4 抽出部 36 は、第 2 抽出部 35 によって抽出された複数のサマリ間パスから、簡易クリティカルサマリ間パスを抽出し、抽出された簡易クリティカルサマリ間パスは、表示制御部 37 により表示部 40 に表示される。さらに、処理部 30 は、簡易クリティカルサマリ間パス上の各サマリで抽出された簡易クリティカルパスを結合し、全体の簡易クリティカルパスを作成する。このように作成された全体の簡易クリティカルパスは、表示制御部 37 により表示部 40 に表示され（ステップ S 100）、処理部 30 は処理を終了する。

【0044】

ついで、図 7 に示すフローチャート（ステップ S 31 ~ S 37）に従って、図 1 に示すスケジュール管理装置 1 によるスケジュールパスの抽出動作（図 6 のステップ S 30 におけるスケジュールパスの抽出処理）について説明する。なお、ステップ S 31 ~ S 36 によって基本パス生成処理が実行されている。

【0045】

まず、処理部 30（第 1 判別部 31）は、格納部 20 のプロセス管理リストから、対象となる全プロセスの終了日（終了予定日）を抽出し、終了日リストを作成する（ステップ S 31）。第 1 判別部 31 は、終了日リストから、終了日の最も早いプロセス（第 1 プロセス）を選択する（ステップ S 32）。そして、第 1 判別部 31 は、格納部 20 のプロセス管理リストを参照し、選択されたプロセスの終了日以降に開始されるプロセス（第 2 プロセス）を抽出する（ステップ S 33）。

【0046】

このとき、第 2 プロセスが抽出された場合（ステップ S 34 の YES ルート）、第 1 判別部 31 は、第 1 プロセスと第 2 プロセスとの間に依存関係が有ると判別し、第 1 プロセスと第 2 プロセスとのペアを基本パスとして格納部 20 に記憶する（ステップ S 35）。第 2 プロセスが抽出されなかった場合（ステップ S 34 の NO ルート）、処理部 30 は、ステップ S 35 をスキップしてステップ S 36 の処理に移行する。

【0047】

ステップ S 36 において、処理部 30 は、終了日リストにおいて全てのプロセスの終了日が選択されたか否か、つまり全てのプロセスの終了日に対しステップ S 32 ~ S 35 の処理が実行されたか否かを判定する。全てのプロセスの終了日が選択されていない場合（ステップ S 36 の NO ルート）、処理部 30 は、ステップ S 32 の処理に戻る。一方、全

10

20

30

40

50

てのプロセスの終了日が選択されている場合（ステップS36のYESルート）、第1抽出部32は、格納部20に記憶された基本パスを、依存線を介して結合し、スケジュールパスを生成し、スケジュールパスのリストを作成する。また、生成されたスケジュールパスは、表示制御部37により表示部40に表示される（ステップS37）。

【0048】

なお、図6のステップS60における、サマリ毎のスケジュールパスの抽出処理も、上述したステップS31～S37と同様の手順で実行される。

また、図6のステップS90におけるサマリ間パスの抽出処理は、上述したステップS31～S37とほぼ同様の手順で実行される。ただし、上述したステップS31～S37における、プロセスおよびスケジュールパスは、それぞれサマリおよびサマリ間パスに置き換えられるとともに、第1判別部31および第1抽出部32は、それぞれ第2判別部34および第3抽出部35に置き換えられる。

【0049】

ついで、図8に示すフローチャート（ステップS41～S44）に従って、図1に示すスケジュール管理装置1による簡易クリティカルパスの抽出動作（図6のステップS40における簡易クリティカルパスの抽出処理）について説明する。

第2抽出部33は、図7のステップS37で作成された、スケジュールパスのリストを参照し、プロセスの段数が最も多いスケジュールパスを抽出する（ステップS41）。そして、第2抽出部33は、ステップS41で複数のスケジュールが抽出されたか否か、つまりプロセスの段数が最も多いスケジュールパスが複数本存在するか否かを判定する（ステップS42）。プロセスの段数が最も多いスケジュールパスが1本だけであれば（ステップS42のNOルート）、第2抽出部33は、ステップS41で抽出されたスケジュールパスを簡易クリティカルパスとして抽出する。抽出された簡易クリティカルパスは、表示制御部37により表示部40に表示される（ステップS43）。

【0050】

一方、プロセスの段数が最も多いスケジュールパスが複数本存在する場合（ステップS42のYESルート）、第2抽出部33は、抽出された複数本の同一段数のスケジュールパスのうち、期間（日数）の最も長いスケジュールパスを簡易クリティカルパスとして抽出する。抽出された簡易クリティカルパスは、表示制御部37により表示部40に表示される（ステップS44）。なお、プロセスの段数も期間（日数）も同じ複数本のスケジュールパスが抽出された場合、第2抽出部33は、当該複数本のスケジュールパスを全て簡易クリティカルパスとして抽出する。

【0051】

なお、図6のステップS70における、サマリ内の簡易クリティカルパスの抽出処理も、上述したステップS41～S44と同様の手順で実行される。

また、図6のステップS100における簡易クリティカルサマリ間パスの抽出処理は、上述したステップS41～S44とほぼ同様の手順で実行される。ただし、上述したステップS41～S44における、プロセス、スケジュールパスおよび簡易クリティカルパスは、それぞれサマリ、サマリ間パスおよび簡易クリティカルサマリ間パスに置き換えられるとともに、第2抽出部33は第4抽出部36に置き換えられる。

【0052】

ここで、図9～図13を参照しながら、図1に示すスケジュール管理装置1の動作を具体的に説明する。つまり、図9に示すようなプロセス管理リストに基づき、サマリIDがSM1であるサマリに含まれる3つのプロセスA～Cから、スケジュールパスおよび簡易クリティカルパスを抽出する具体的な手順について説明する。つまり、ここで説明する手順は、サマリ内のプロセスについてスケジュールパスおよび簡易クリティカルパスを抽出する手順であり、図6のステップS60、S70での処理に対応する。

【0053】

図9に示すプロセス管理リストから得られる、サマリSM1におけるプロセスの終了日リストは、プロセスAの2011/6/26と、プロセスBの2011/7/1と、プロセスCの2011/7/7

10

20

30

40

50

とを含む。第1判別部31は、終了日の最も早いプロセスAを選択し(図7のステップS32;図10の「スタート」参照)、図9に示すプロセス管理リストを参照し、プロセスAの終了日以降に開始される2つのプロセスB,Cを抽出する(図7のステップS33;図9,図11参照)。したがって、第1判別部31は、プロセスAとプロセスBとの間に依存関係があると判別するとともに、プロセスAとプロセスCとの間にも依存関係があると判別し、プロセスA~BとプロセスA~Cとの2組を基本パスとして格納部20に格納する(図7のステップS35)。

【0054】

この後、第1判別部31は、終了日の次に早いプロセスBを選択し(図7のステップS32;図12の「スタート」参照)、図9に示すプロセス管理リストを参照し、プロセスBの終了日以降に開始されるプロセスCを抽出する(図7のステップS33;図12参照)。したがって、第1判別部31は、プロセスBとプロセスCとの間に依存関係があると判別し、プロセスB~Cを基本パスとして格納部20に格納する(図7のステップS35)。これにより、サマリSM1内におけるプロセスについて、全ての基本パス(3本の基本パス)が抽出される。

10

【0055】

ついで、第1抽出部32は、格納部20に記憶された基本パスを、依存線を介して結合し、スケジュールパスを生成する。図12に示す例では、2組の基本パスつまりプロセスA~BとプロセスB~Cとを依存線を介して結合した、一つ目のスケジュールパス(プロセスA~プロセスB~プロセスC)が生成される。また、図12に示す例では、基本パスであるプロセスA~Cが、二つ目のスケジュールパス(プロセスA~プロセスC)として生成される。したがって、一つ目のスケジュールパスにおけるプロセスの接続段数は3であり、二つ目のスケジュールパスにおけるプロセスの接続段数は2である。

20

【0056】

そして、第2抽出部33は、図12に示す2つスケジュールパスのうち、プロセスの接続段数が多い一つ目のスケジュールパス(プロセスA~プロセスB~プロセスC)を、図13に示すように、簡易クリティカルパスとして抽出する。

他のサマリにおけるプロセスについても、上述と同様にしてスケジュールパスおよび簡易クリティカルパスが抽出される。

30

【0057】

また、複数のサマリ相互間についても、上述と同様にしてサマリ間パスおよび簡易クリティカルサマリ間パスが抽出される。そして、簡易クリティカルサマリ間パス上の各サマリで抽出された簡易クリティカルパスが結合され、全体の簡易クリティカルパスが作成され、表示部40に表示される。

【0058】

〔1-3〕第1実施形態のスケジュール管理装置の作用効果

第1実施形態のスケジュール管理装置1によれば、オペレータ等は、従来の複雑な要素の全てを入力せず、各プロセスの開始予定タイミングと終了予定タイミングとを登録するだけでスケジュールパスおよび簡易クリティカルパスを抽出することができる。つまり、各プロセスの開始予定タイミングおよび終了予定タイミングの前後関係や間隔に基づきプロセス相互間の依存関係が判別され、相互に依存関係があるプロセスが数珠つなぎにされて、予定される一連のプロセス(スケジュールパス)が作成される。また、複数のスケジュールパスの中で、プロセスの接続段数の多さ(構成の複雑さ)やスケジュールパスの期間(日数)が比較され、プロジェクト(スケジュール)に影響を与えるであろう重要なパスが、簡易クリティカルパスとして簡易的に抽出され、表示部40に表示される。

40

【0059】

これにより、厳密な意味でのクリティカルパスではないが設計におけるキーとなりうるスケジュールパスが、簡易クリティカルパスとして簡易的に各プロセスの開始予定タイミングおよび終了予定タイミングに基づき抽出され表示される。つまり、少ない情報(プロセスの内容まで考慮する必要無し)でプロセス相互間の依存関係を判別するため、簡易的

50

にプロセスのスケジュールパスを抽出できるとともに、妥当性の高い情報に基づいて簡易クリティカルパスを判別するため、判別精度も高い。

【0060】

したがって、プロセスの前後関係の精度を大きく崩すことなく、プロセスのスケジュールパスを簡易的に作成して簡易クリティカルパスを抽出表示することができ、簡易クリティカルパスを設計マネジメント材料として活用することができる。つまり、スケジュールパスの表示の重要要素となっている各プロセスの期間の情報を活用しスケジュールパスおよび簡易クリティカルパスを簡易的に生成することで、短期間での作業見積りやスケジュールの見直しを実施できる環境が整えられる。

【0061】

また、利用者（オペレータ等）は、従来の複雑な要素の全てを入力せず、各プロセスの開始予定タイミングと終了予定タイミングとを登録するだけでよいため、利用者（オペレータ等）の入力負担が大幅に軽減され、入力登録作業を短期間で行なってスケジュールパスや簡易クリティカルパスを得ることができる。さらに、入力ミスが発生しやすかった複雑な入力登録作業（成果物相互間の入出力関係やプロセス相互間の依存関係などの入力）を行なう必要がなくなるため、入力ミスを抑止することができる。また、入力登録作業を短期間で行なえるので、スケジュール変更の度に入力登録作業が発生しても、プロジェクトを円滑に運用することができる。

【0062】

また、サマリについても、スケジュールパスや簡易クリティカルパスと同様、サマリ間パスや簡易クリティカルサマリ間パスを抽出して表示することができ、上述と同様の作用効果を得ることができる。さらに、簡易クリティカルサマリ間パス上の各サマリで抽出された簡易クリティカルパスを結合することで、全体の簡易クリティカルパスを作成することができる。また、サマリ単位でプロセスを分割し各サマリ内のプロセスについてスケジュールパスや簡易クリティカルパスを抽出して保存しておくことで、変更のあったプロセスを含むサマリについて再抽出処理を行なうだけで当該変更に対応することが可能になる。

【0063】

以上のように、第1実施形態のスケジュール管理装置1によれば、依存関係判断に必要であった全要素（上記項目(1)～(5)参照）を入力することなく開始予定タイミングおよび終了予定タイミングだけでスケジュールパスや簡易クリティカルパスが推測表示される。これにより、短時間の入力登録作業でスケジュールパスや簡易クリティカルパスを抽出表示することが可能になり、注目するパスを分かりやすく表現しスケジュールの見直しなどを判断材料とすることが可能になる。このような効果は、スケジュール構築初期段階の確認作業や短納期のプロジェクトにおいて有効になる。

【0064】

以下に、第1実施形態のスケジュール管理装置1によって得られる作用効果、特にスケジュール登録作業の効率化について、具体的に説明する。

プロジェクト立ち上げ時のスケジュール作成や、進捗状況に合わせてのスケジュールの見直しによる依存関係の作成変更は、プロジェクト期間中、月一や隔週などで定期的発生する。ここで、一回のスケジュールの見直しや作成にかかる時間について、厳密なクリティカルパスを抽出する場合と、本実施形態のごとく簡易クリティカルパスを抽出する場合とを比較して示す。

【0065】

厳密（詳細）なクリティカルパスを抽出する場合の1プロセス分の登録作業に際し、各入力対象（要素）1件当たりに要する登録時間は、表1の通りである。

【0066】

10

20

30

40

【表 1】

入力対象	1件当たりの登録時間
開始予定日	3 秒
終了予定日	3 秒
優先順位	3 秒
リスク対象フラグ	3 秒
プロセス間依存関係	10 秒 × 依存数
担当者(リソース)間の依存関係	10 秒 × 依存数
成果物間の依存関係	10 秒 × 依存数

10

## 【 0 0 6 7 】

したがって、依存数を 2 とした場合、1 プロセスあたりに要する登録時間は、7 2 秒となる。平均的なプロジェクトのプロセス数を 1 0 0 とした場合、1 プロジェクトあたりに要する登録時間は、7 2 0 0 秒となる。実際には、上記登録時間のほかに依存関係の構成検討に数時間を要し、この検討時間は、例えば 1 プロジェクト当たり 5 時間程度になる。

一方、本実施形態のごとく簡易クリティカルパスを抽出する場合、1 プロセス分の登録作業に際し、各入力対象(要素) 1 件あたりに要する登録時間は、表 2 の通りである。

## 【 0 0 6 8 】

20

【表 2】

入力対象	1件当たりの登録時間
開始予定日	3 秒
終了予定日	3 秒

## 【 0 0 6 9 】

したがって、1 プロセスあたりに要する登録時間は、6 秒となる。平均的なプロジェクトのプロセス数を 1 0 0 とした場合、1 プロジェクトあたりに要する登録時間は、6 0 0 秒となる。実際には、上記登録時間のほかに依存関係の構成検討に数時間を要するが、この場合、期間(開始予定日/終了予定日)の検討のみになるため、その検討時間は、例えば 1 プロジェクト当たり 1 時間程度になる。

30

## 【 0 0 7 0 】

これにより、本実施形態によれば、一回のスケジュールの見直しや作成にかかる時間は、以下の通り、1 プロジェクト当たり、登録作業時間で 1 時間 5 0 分短縮され、検討時間で 4 時間短縮され、合計で 5 時間 5 0 分短縮され、スケジュール登録作業が大幅に効率化される。なお、ここでは、プロジェクト全体の構成などの要素の検討時間は除いている。

登録作業時間の短縮時間：7 2 0 0 秒 - 6 0 0 秒 = 6 6 0 0 秒(1 時間 5 0 分)

検討時間の短縮時間：5 時間 - 1 時間 = 4 時間

合計の短縮時間：1 時間 5 0 分 + 4 時間 = 5 時間 5 0 分

40

## 【 0 0 7 1 】

## 〔 2 〕 第 2 実施形態

## 〔 2 - 1 〕 第 2 実施形態のスケジュール管理装置の構成および機能

図 1 4 は、第 2 実施形態のスケジュール管理装置 1 A のハードウェア構成および機能構成を示すブロック図である。なお、図中、既述の符号と同一の符号は、同一もしくはほぼ同一の部分を示しているため、その詳細な説明は省略する。

## 【 0 0 7 2 】

図 1 4 に示すスケジュール管理装置 1 A は、第 1 実施形態のスケジュール管理装置 1 と同様、バス 5 0 経由で相互に通信可能に接続された登録部 1 0 , 格納部 2 0 , 処理部 3 0 および表示部 4 0 を有している。ただし、スケジュール管理装置 1 A では、格納部 2 0 に

50

は図15に示すようなプロセス管理リストが保存されるとともに、処理部30は、格納部20に記憶されたスケジュール管理プログラムを読み出して実行することで、上述した第1判別部31，第1抽出部32，第2抽出部33，第2判別部34，第3抽出部35，第4抽出部36および表示制御部37としての機能を果たすほか、差分算出部38およびシフト処理部39としての機能を果たす。なお、差分算出部38およびシフト処理部39は、パス調整部30cを構成している。

【0073】

また、第2実施形態において、オペレータ等は、登録部10から、作業の進行に伴い実際に開始または終了したプロセスについて実開始タイミング（開始日（実績））または実終了タイミング（終了日（実績））を格納部20のプロセス管理リストに登録する。

10

【0074】

さらに、第2実施形態では、図15に示すようなプロセス管理リストが用いられる。なお、図15は、図14に示すスケジュール管理装置1Aで用いられるプロセス管理リストの例を示す図である。図15に示すプロセス管理リストでは、図3に示す第1実施形態のプロセス管理リストに加え、開始日（実績），終了日（実績）および日数（実績）の欄が追加されている。これらの欄は、オペレータ等によって実際に開始または終了した開始日（実績）/終了（実績）に登録されるまで空欄になっている。なお、図15における「開始日（予定）」および「終了日（予定）」は、それぞれ第1実施形態における「開始日」および「終了日」に対応している。

【0075】

開始日（実績）は、オペレータ等によって登録される、実際にプロセスを開始した開始日（年月日）であり、終了日（実績）は、オペレータ等によって登録される、実際にプロセスを終了した終了日（年月日）である。また、日数（実績）は、開始日（実績）と終了日（実績）とに基づき自動算出されるもので、各プロセスの実行に実際に要した期間である。なお、本実施形態では、実開始タイミング/実終了タイミングを年月日として登録し、各プロセスの期間（実績）を日数として登録しているが、実開始タイミング/実終了タイミングを実開始日時/実終了日時として登録し、各プロセスの期間（実績）を時間単位または分単位として登録してもよい。

20

【0076】

差分算出部38は、作業の進行に伴い開始または終了したプロセスについて開始日（実績）または終了日（実績）が格納部20のプロセス管理リストに登録された場合、以下の処理を実行する。つまり、差分算出部38は、開始または終了したプロセスの属するスケジュールパスについて、開始日（実績）と当該開始日（実績）に対応する開始日（予定）との差分、または、終了日（実績）と当該終了日（実績）に対応する終了日（予定）との差分を算出する。

30

【0077】

シフト処理部39は、開始または終了したプロセスの属するスケジュールパスについて、開始または終了したプロセス以降のプロセスの開始日（予定）および終了日（予定）タイミングを、差分算出部38によって算出された差分だけシフトする。

シフト処理部39によるシフトを行なった後、第2抽出部33は、第1判別部31および第1抽出部31と協働して、開始または終了したプロセス以降のプロセスについて、再度、スケジュールパスおよび簡易クリティカルパスを抽出する。

40

【0078】

〔2-2〕第2実施形態のスケジュール管理装置の動作

次に、図16に示すフローチャート（ステップS110～S180）に従って、図17～図22を参照しながら、図14に示すスケジュール管理装置1Aの動作について説明する。なお、図17～図22は、図14に示すスケジュール管理装置1Aの動作を具体的に説明する図である。

【0079】

ここでは、まず、第1実施形態のスケジュール管理装置1と同様の処理により、図17

50

に示すような、プロセス管理リスト（図17の左側参照）と、スケジュールパスおよび簡易クリティカルパス（図17の右側参照）とが得られているものとする。図17のガントチャートでは、2本のスケジュールパス、つまり、プロセス1, 2, 3を依存線で接続したパスと、プロセス4, 5, 6を依存線で接続したパスとが表示されるとともに、これら2本のスケジュールパスのうち、プロセス4, 5, 6を依存線で接続したパスが簡易クリティカルパスとして表示されている。

【0080】

上述した状態で、処理部30は、オペレータ等により登録部10から実績（開始日（実績）および/または終了日（実績））が登録されたか否かを判定する（ステップS110）。実績が登録された場合（ステップS110のYESルート）、例えば図18に示すようにプロセス1, 4について開始日（実績）および終了日（実績）が登録された場合、処理部30は、実績登録のあったプロセス（実際に開始または終了したプロセス）の属するスケジュールパスを確認する（ステップS120）。そして、処理部30は、当該スケジュールパスにおいて、実績登録のあったプロセスの下位に未実施のプロセスがあるか否かを判定する（ステップS130）。

10

【0081】

未実施のプロセスが無い場合（ステップS130のNOルート）、処理部30は、ステップS110の処理に戻る。

一方、未実施のプロセスがある場合（ステップS130のYESルート）、差分算出部38は、実績登録のあったプロセスの属するスケジュールパスについて、開始日（実績）と当該開始日（実績）に対応する開始日（予定）との差分、または、終了日（実績）と当該終了日（実績）に対応する終了日（予定）との差分を算出する（ステップS140）。

20

【0082】

図18に示す例では、差分算出部38は、プロセス1について登録された終了日（実績）2001/6/27と、この終了日（実績）に対応する終了日（予定）2011/6/25との差分（2日遅延）を算出する。同様に、差分算出部38は、プロセス4について登録された終了日（実績）2001/6/27と、この終了日（実績）に対応する終了日（予定）2011/6/27との差分（差分なし）を算出する。

【0083】

そして、処理部30は、差分が無い場合（例えば図18のプロセス4の場合；ステップS150のNOルート）、処理を終了する。一方、差分が有る場合（例えば図18のプロセス1の場合；ステップS150のYESルート）、シフト処理部39は、実績登録のあったプロセスの属するスケジュールパスについて、実績登録のあったプロセスの下位における未実施のプロセスの開始日（予定）および終了日（予定）を、差分だけシフトする（ステップS160）。シフト処理部39は、図19に示すように、シフト後の情報（未実施のプロセスの開始日（予定）および終了日（予定））を、格納部20のプロセス管理リストに登録する（ステップS170）。

30

【0084】

図18に示す例では、プロセス1について2日の遅延が生じているので、シフト処理部39は、プロセス1の属するスケジュールパスにおける、プロセス1よりも下位の未実施のプロセス2, 3の開始日（予定）および終了日（予定）を2日だけ遅延シフトする。つまり、図19に示すように、プロセス2の開始日（予定）2011/6/26および終了日（予定）2011/6/28は、それぞれ開始日（予定）2011/6/28および終了日（予定）2011/6/30にシフトされ、プロセス3の開始日（予定）2011/6/29および終了日（予定）2011/7/2は、それぞれ開始日（予定）2011/7/1および終了日（予定）2011/7/4にシフトされる。なお、プロセス4については差分が0であり、プロセスの遅延も前倒しも生じておらず、シフト処理部39による処理は何ら行なわれない（ステップS150のNOルート）。

40

【0085】

なお、図20（A）に示すように、あるプロセスについて開始日（実績）および終了日（実績）の両方が登録されている場合、終了日（実績）と、当該終了日（実績）に対応す

50

る終了日（予定）との差分が算出される。そして、図20（B）に示すように、1日の遅延が生じている場合、当該プロセスに続く他のプロセスの開始日（予定）および終了日（予定）が1日ずつ遅延シフトされる。

【0086】

また、図21（A）に示すように、あるプロセスについて開始日（実績）のみが登録されている場合、開始日（実績）と、当該開始日（実績）に対応する開始日（予定）との差分が算出との差分が算出される。そして、図21（B）に示すように、1日の前倒しが生じている場合、当該プロセスの終了日（予定）が1日だけ前倒しシフトされるとともに、当該プロセスに続く他のプロセスの開始日（予定）および終了日（予定）が1日ずつ前倒しシフトされる。

10

【0087】

この後、実績登録のあったプロセス以降のプロセスについて、第2抽出部33は、第1判別部31および第1抽出部31と協働して、再度、スケジュールパスおよび簡易クリティカルパスを抽出する処理（図6のステップS20～S100の処理）を実行する（ステップS180）。

【0088】

図19に示す例では、実績登録のあったプロセス1の属するスケジュールパスにおけるシフト処理後（パス調整後）の未実施のプロセス2, 3と、他のスケジュールパスにおける未実施のプロセス5, 6とについて、再度、スケジュールパスおよび簡易クリティカルパスを抽出する処理が実行される。これにより、例えば図22に示すように、2本のスケジュールパス、つまり、プロセス2, 3を依存線で接続したパスと、プロセス5, 6を依存線で接続したパスとが表示されるとともに、これら2本のスケジュールパスのうち、プロセス2, 3を依存線で接続したパスが簡易クリティカルパスとして表示される。このように、図17～図19および図22に示す例では、当初、プロセス4, 5, 6を依存線で接続したパスが簡易クリティカルパスとして表示されていたが、プロセス1の実績を反映させることで、プロセス2, 3のパスが簡易クリティカルパスとして表示されることになる。

20

【0089】

〔2-3〕第2実施形態のスケジュール管理装置の作用効果

第2実施形態のスケジュール管理装置1Aによれば、第1実施形態のスケジュール管理装置1と同様の作用効果が得られるほか、プロセスを実際に行うことにより得られる実績の遅延期間/前倒し期間に応じて、プロセスの開始タイミング/終了タイミングがシフトされ更新される。これにより、スケジュールの変更が容易に行なわれる。また、シフト後のスケジュール（プロセス管理リスト）に従って、スケジュールパスや簡易クリティカルパスの再抽出および再表示が自動的に行なわれるので、オペレータ等は、スケジュールの見直しを容易かつ確実に行なうことができる。

30

【0090】

〔3〕その他

以上、本発明の好ましい実施形態について詳述したが、本発明は、係る特定の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲内において、種々の変形、変更して実施することができる。

40

【0091】

上述した第1判別部31, 第1抽出部32, 第2抽出部33, 第2判別部34, 第3抽出部35, 第4抽出部36, 表示制御部37, 差分算出部38およびシフト処理部39としての機能の全部もしくは一部は、コンピュータ（CPU, プロセッサ, 情報処理装置, 各種端末を含む）が所定のアプリケーションプログラム（スケジュール管理プログラム）を実行することによって実現される。

【0092】

そのプログラムは、例えばフレキシブルディスク, CD（CD-ROM, CD-R, CD-RWなど）, DVD（DVD-ROM, DVD-RAM, DVD-R, DVD-RW

50

、DVD+R、DVD+RWなど)、ブルーレイディスク等のコンピュータ読取可能な記録媒体に記録された形態で提供される。この場合、コンピュータはその記録媒体からプログラムを読み取って内部記憶装置または外部記憶装置に転送し格納して用いる。

【0093】

ここで、コンピュータとは、ハードウェアとOS(オペレーティングシステム)とを含む概念であり、OSの制御の下で動作するハードウェアを意味している。また、OSが不要でアプリケーションプログラム単独でハードウェアを動作させるような場合には、そのハードウェア自体がコンピュータに相当する。ハードウェアは、少なくとも、CPU等のマイクロプロセッサと、記録媒体に記録されたコンピュータプログラムを読み取る手段とをそなえている。上記スケジュール管理プログラムは、上述のようなコンピュータに、第1判別部31、第1抽出部32、第2抽出部33、第2判別部34、第3抽出部35、第4抽出部36、表示制御部37、差分算出部38およびシフト処理部39としての機能を実現させるプログラムコードを含んでいる。また、その機能の一部は、アプリケーションプログラムではなくOSによって実現されてもよい。

10

【0094】

(4)付記

以上の実施形態に関し、さらに以下の付記を開示する。

(付記1)

複数のプロセスを含む作業のスケジュールを管理するコンピュータに、前記複数のプロセスのそれぞれについて予め登録された開始予定タイミングおよび終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、前記複数のプロセスのうち、前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、処理を実行させる、スケジュール管理プログラム。

20

【0095】

(付記2)

前記複数のプロセスのうちの判別対象の第1プロセスと開始予定タイミングが当該第1プロセスの終了予定タイミング以降である第2プロセスとの間に前記依存関係が有ると判別する、

処理を前記コンピュータに実行させる、付記1に記載のスケジュール管理プログラム。

30

【0096】

(付記3)

抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する、

処理を前記コンピュータに実行させる、付記1または付記2に記載のスケジュール管理プログラム。

【0097】

(付記4)

接続されたプロセスの段数が最も多いスケジュールパスを前記簡易クリティカルパスとして抽出する、

処理を前記コンピュータに実行させる、付記3に記載のスケジュール管理プログラム。

40

【0098】

(付記5)

前記接続されたプロセスの段数が最も多いスケジュールパスが複数本抽出された場合、当該複数本の同一段数のスケジュールパスのうち、期間の最も長いスケジュールパスを前記簡易クリティカルパスとして抽出する、

処理を前記コンピュータに実行させる、付記4に記載のスケジュール管理プログラム。

【0099】

(付記6)

前記複数のプロセスが複数のサマリにグループ化されている場合、各サマリに含まれる

50

前記 2 以上のプロセスについて、前記簡易クリティカルパスを抽出し、

前記複数のサマリのそれぞれについて、各サマリに含まれる 2 以上のプロセスについて予め登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のサマリ相互間の依存関係の有無を判別し、

前記複数のサマリのうち、前記依存関係が有ると判別されたサマリどうしを接続して複数のサマリ間パスを抽出し、

抽出された前記複数のサマリ間パスのうち、サマリ構成の複雑なサマリ間パスを簡易クリティカルサマリ間パスとして抽出する、

処理を前記コンピュータに実行させる、付記 3 ~ 付記 5 のいずれか一項に記載のスケジュール管理プログラム。

【 0 1 0 0 】

( 付記 7 )

接続されたサマリの段数が最も多いサマリ間パスを前記簡易クリティカルサマリ間パスとして抽出する一方、

前記接続されたサマリの段数が最も多いサマリ間パスが複数本抽出された場合、当該複数本の同一段数のサマリ間パスのうち、期間の最も長いサマリ間パスを前記簡易クリティカルサマリ間パスとして抽出する、

処理を前記コンピュータに実行させる、付記 6 に記載のスケジュール管理プログラム。

【 0 1 0 1 】

( 付記 8 )

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出し、

前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、算出された前記差分だけシフトする、

処理を前記コンピュータに実行させる、付記 1 ~ 付記 7 のいずれか一項に記載のスケジュール管理プログラム。

【 0 1 0 2 】

( 付記 9 )

前記開始予定タイミングおよび前記終了予定タイミングを前記差分だけシフトした後、前記開始または終了したプロセス以降のプロセスについて、再度、前記簡易クリティカルパスを抽出する、

処理を前記コンピュータに実行させる、付記 8 に記載のスケジュール管理プログラム。

【 0 1 0 3 】

( 付記 1 0 )

複数のプロセスを含む作業のスケジュールを管理するスケジュール管理装置であって、前記複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミングを予め登録される格納部と、

前記格納部に登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別する第 1 判別部と、

前記複数のプロセスのうち、前記第 1 判別部によって前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する第 1 抽出部と、を有する、スケジュール管理装置。

【 0 1 0 4 】

( 付記 1 1 )

前記第 1 判別部は、前記複数のプロセスのうちの判別対象の第 1 プロセスと開始予定タイミングが当該第 1 プロセスの終了予定タイミング以降である第 2 プロセスとの間に前記

10

20

30

40

50

依存関係が有ると判別する、付記 10 に記載のスケジュール管理装置。

【0105】

(付記 12)

前記第 1 抽出部によって抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する第 2 抽出部、をさらに有する、付記 11 に記載のスケジュール管理装置。

【0106】

(付記 13)

前記第 2 抽出部は、接続されたプロセスの段数が最も多いスケジュールパスを前記簡易クリティカルパスとして抽出する、付記 12 に記載のスケジュール管理装置。

10

【0107】

(付記 14)

前記第 2 抽出部は、前記接続されたプロセスの段数が最も多いスケジュールパスが複数本抽出された場合、当該複数本の同一段数のスケジュールパスのうち、期間の最も長いスケジュールパスを前記簡易クリティカルパスとして抽出する、付記 13 に記載のスケジュール管理装置。

【0108】

(付記 15)

前記複数のプロセスが複数のサマりにグループ化されている場合、前記第 2 抽出部は、各サマりに含まれる前記 2 以上のプロセスについて、前記簡易クリティカルパスを抽出し

20

、  
前記複数のサマリのそれぞれについて、各サマりに含まれる 2 以上のプロセスについて前記格納部 20 に予め登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のサマリ相互間の依存関係の有無を判別する第 2 判別部と、

前記複数のサマリのうち、前記第 2 判別部によって前記依存関係が有ると判別されたサマリどうしを接続して複数のサマリ間パスを抽出する第 3 抽出部と、

前記第 3 抽出部によって抽出された前記複数のサマリ間パスのうち、サマリ構成の複雑なサマリ間パスを簡易クリティカルサマリ間パスとして抽出する第 4 抽出部と、さらに有する、付記 12 ~ 付記 14 のいずれか一項に記載のスケジュール管理装置。

【0109】

(付記 16)

前記第 4 抽出部は、

接続されたサマリの段数が最も多いサマリ間パスを前記簡易クリティカルサマリ間パスとして抽出する一方、

前記接続されたサマリの段数が最も多いサマリ間パスが複数本抽出された場合、当該複数本の同一段数のサマリ間パスのうち、期間の最も長いサマリ間パスを前記簡易クリティカルサマリ間パスとして抽出する、付記 15 に記載のスケジュール管理装置。

【0110】

(付記 17)

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが前記格納部に登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出する差分算出部と、

40

前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、前記差分算出部によって算出された前記差分だけシフトするシフト処理部と、をさらに有する、付記 10 ~ 付記 16 のいずれか一項に記載のスケジュール管理装置。

【0111】

50

(付記 18)

前記シフト処理部によって前記開始予定タイミングおよび前記終了予定タイミングを前記差分だけシフトした後、前記第2抽出部は、前記開始または終了したプロセス以降のプロセスについて、再度、前記簡易クリティカルパスを抽出する、付記17に記載のスケジュール管理装置。

【0112】

(付記 19)

複数のプロセスを含む作業のスケジュールを管理するスケジュール管理方法であって、前記複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミングを予め格納部に登録し、

10

前記格納部に登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、

前記複数のプロセスのうち、前記依存関係があると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、スケジュール管理方法。

【0113】

(付記 20)

抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する、付記19に記載のスケジュール管理方法。

【0114】

(付記 21)

接続されたプロセスの段数が最も多いスケジュールパスを前記簡易クリティカルパスとして抽出する、付記20に記載のスケジュール管理方法。

20

【0115】

(付記 22)

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出し、

前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、算出された前記差分だけシフトする、付記19～付記21のいずれか一項に記載のスケジュール管理方法。

30

【0116】

(付記 23)

複数のプロセスを含む作業のスケジュールを管理するコンピュータに、前記複数のプロセスのそれぞれについて予め登録された開始予定タイミングおよび終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、

前記複数のプロセスのうち、前記依存関係があると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、

40

処理を実行させる、スケジュール管理プログラムを記録したコンピュータ読取可能な記録媒体。

【0117】

(付記 24)

抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する、

処理を前記コンピュータに実行させる、付記23に記載のスケジュール管理プログラムを記録したコンピュータ読取可能な記録媒体。

【0118】

(付記 25)

50

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出し、

前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、算出された前記差分だけシフトする、

処理を前記コンピュータに実行させる、付記 2 3 または付記 2 4 のいずれか一項に記載のスケジュール管理プログラムを記録したコンピュータ読取可能な記録媒体。

10

【0119】

(付記 2 6)

複数のプロセスを含む作業のスケジュールを管理するスケジュール管理装置であって、前記複数のプロセスのそれぞれについて開始予定タイミングおよび終了予定タイミングを予め登録される格納部と、

プロセッサと、を有し、

前記プロセッサは、

前記格納部に登録された前記開始予定タイミングおよび前記終了予定タイミングに基づき、前記複数のプロセス相互間の依存関係の有無を判別し、

前記複数のプロセスのうち、前記依存関係が有ると判別されたプロセスどうしを接続して複数のスケジュールパスを抽出する、スケジュール管理装置。

20

【0120】

(付記 2 7)

前記プロセッサは、抽出された前記複数のスケジュールパスのうち、プロセス構成の複雑なスケジュールパスを簡易クリティカルパスとして抽出する、付記 2 6 に記載のスケジュール管理装置。

【0121】

(付記 2 8)

前記プロセッサは、

前記複数のプロセスを含む前記作業の進行に伴い開始または終了したプロセスについて実開始タイミングまたは実終了タイミングが登録された場合、前記開始または終了したプロセスの属する前記スケジュールパスについて、前記実開始タイミングと当該実開始タイミングに対応する前記開始予定タイミングとの差分、または、前記実終了タイミングと当該実終了タイミングに対応する前記終了予定タイミングとの差分を算出し、

30

前記開始または終了したプロセスの属する前記スケジュールパスについて、前記開始または終了したプロセス以降のプロセスの前記開始予定タイミングおよび前記終了予定タイミングを、算出された前記差分だけシフトする、付記 2 6 または付記 2 7 のいずれか一項に記載のスケジュール管理装置。

【符号の説明】

【0122】

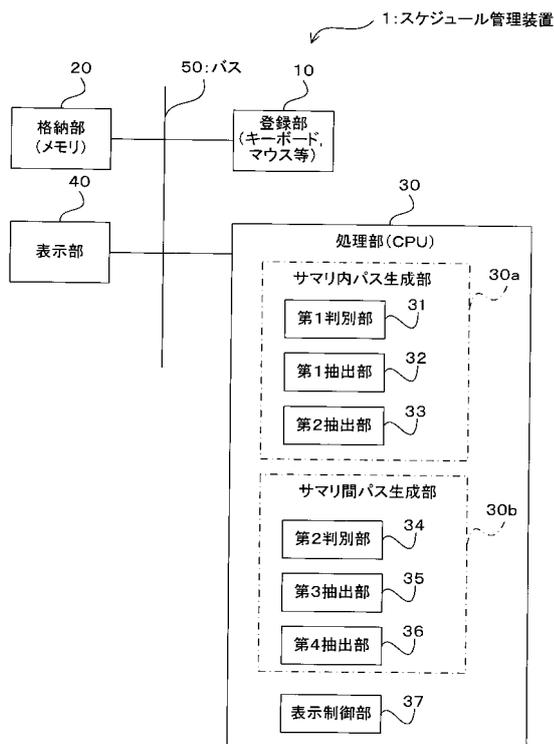
40

- 1, 1 A スケジュール管理装置
- 1 0 登録部 (キーボード, マウス等)
- 2 0 格納部 (メモリ)
- 3 0 処理部 (CPU, コンピュータ, プロセッサ)
- 3 0 a サマリ内パス生成部
- 3 0 b サマリ間パス生成部
- 3 0 c パス調整部
- 3 1 第 1 判別部 (サマリ内パス生成部)
- 3 2 第 1 抽出部 (サマリ内パス生成部)
- 3 3 第 2 抽出部 (サマリ内パス生成部)

50

- 3 4 第2判別部 (サマリ間パス生成部)
- 3 5 第3抽出部 (サマリ間パス生成部)
- 3 6 第4抽出部 (サマリ間パス生成部)
- 3 7 表示制御部
- 3 8 差分算出部 (パス調整部)
- 3 9 シフト処理部 (パス調整部)
- 4 0 表示部
- 5 0 バス

【 図 1 】



【 図 2 】

開始・完了日設定

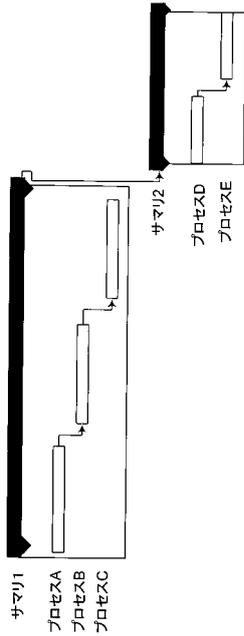
プロセス名	開始予定日	終了予定日
プロセス1	2011/06/24	2011/06/26
プロセス2	2011/06/28	2011/07/01
プロセス3	2011/07/01	2011/07/05
プロセスn	2011/06/01	2011/06/01

登録    キャンセル

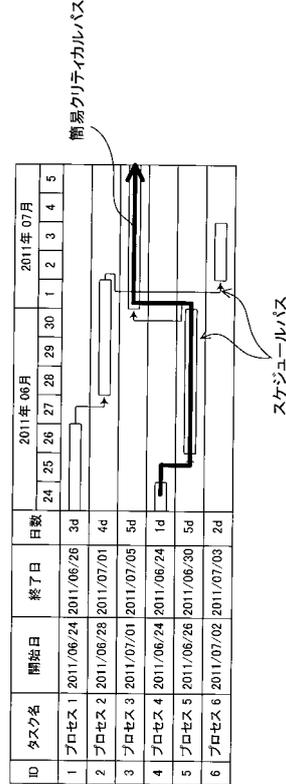
【 図 3 】

ID	工程名	開始日	終了日	日数	サマリID	依存工程ID
1	プロセス1	2011/6/24	2011/6/26	3d	SM1	2
2	プロセス2	2011/6/28	2011/7/1	4d	SM1	
n	プロセスn	2011/12/28	2011/12/31	4d	SM2	

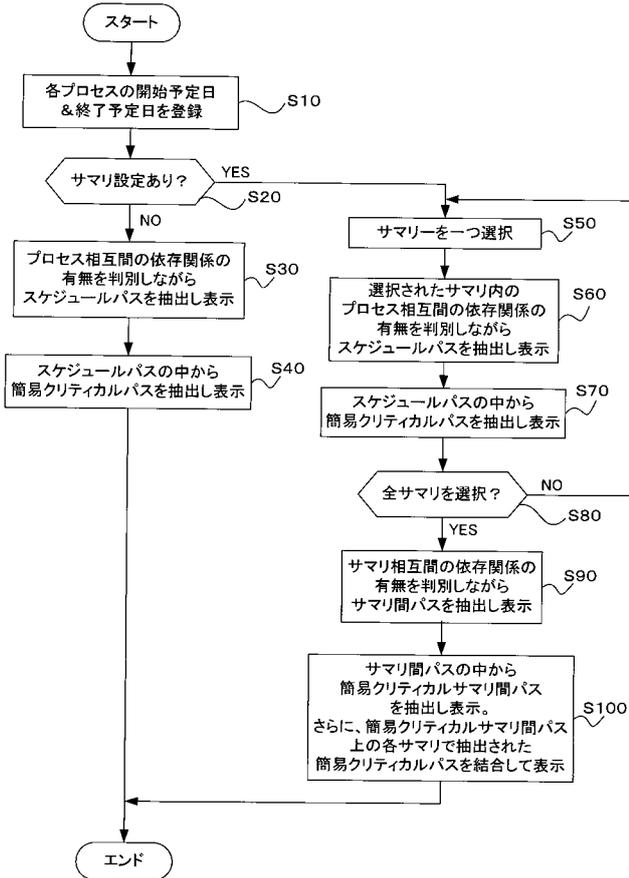
【 図 4 】



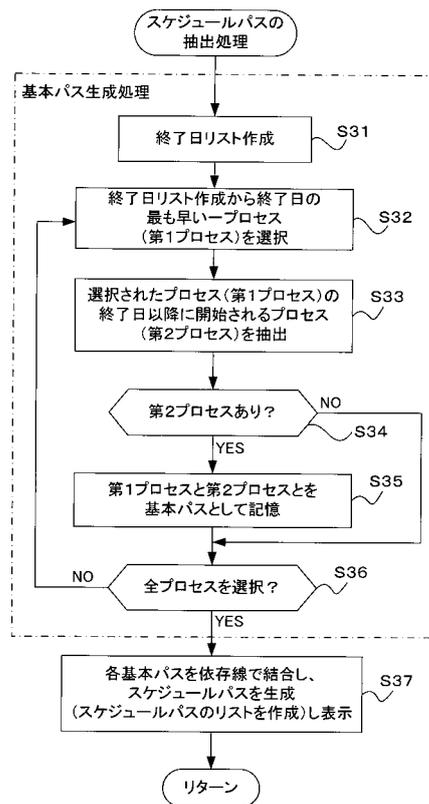
【 図 5 】



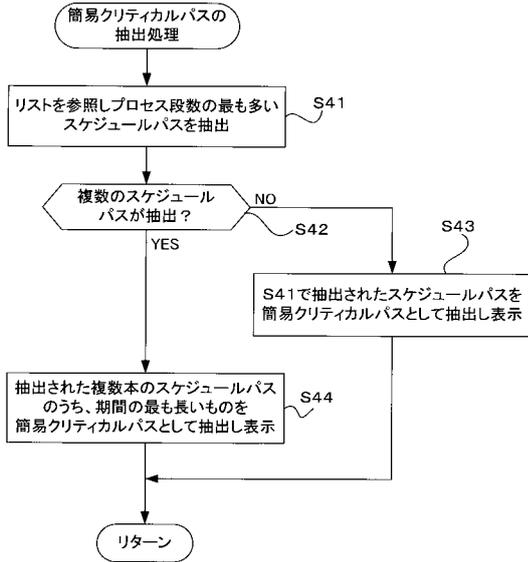
【 図 6 】



【 図 7 】



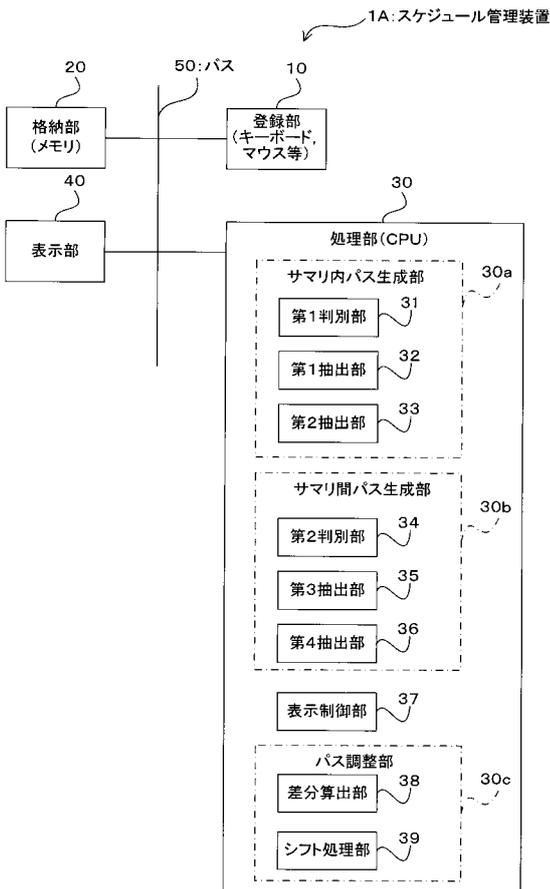
【図 8】



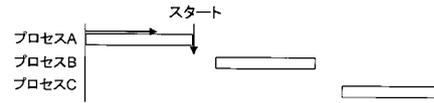
【図 9】

ID	工程名	開始日	終了日	日数	サマリID	依存工程ID
1	プロセスA	2011/6/24	2011/6/26	3d	SM1	
2	プロセスB	2011/6/28	2011/07/01	4d	SM1	
3	プロセスC	2011/07/03	2011/07/07	5d	SM1	
X	プロセスX	2011/12/28	2011/12/31	4d	SM2	

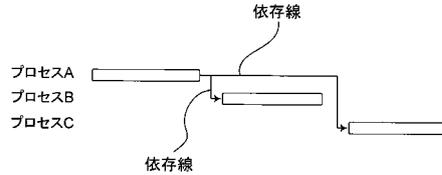
【図 14】



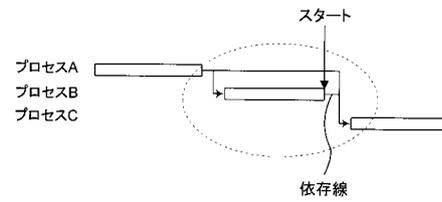
【図 10】



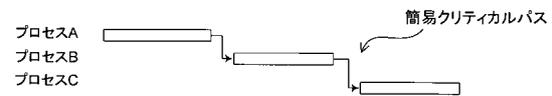
【図 11】



【図 12】



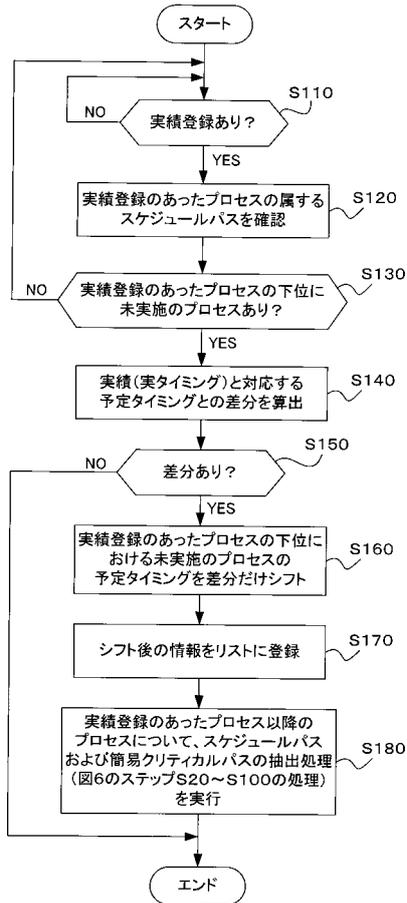
【図 13】



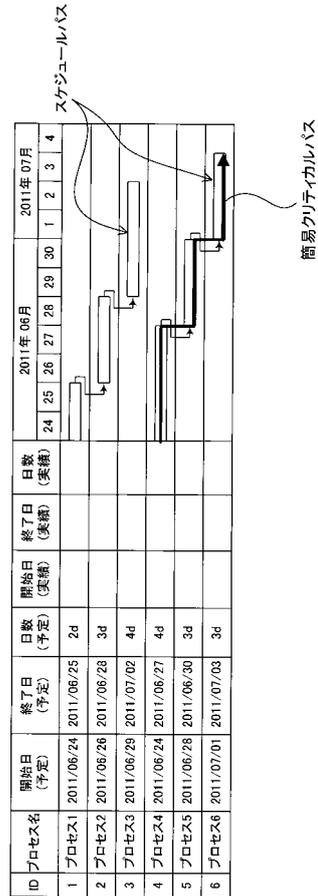
【図 15】

ID	工程名	開始日 (予定)	終了日 (予定)	日数 (予定)	開始日 (実績)	終了日 (実績)	日数 (実績)	サマリID	依存工程ID
1	プロセス1	2011/6/24	2011/6/26	3d	2011/6/24	2011/6/26	3d	SM1	2
2	プロセス2	2011/6/28	2011/7/1	4d				SM1	
n	プロセスn	2011/12/28	2011/12/31	4d				SM2	

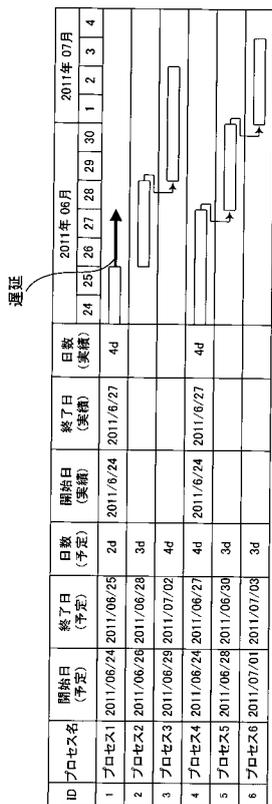
【 図 1 6 】



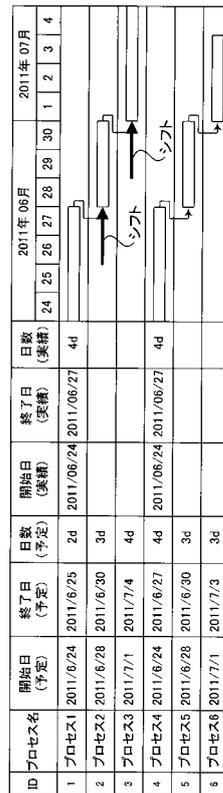
【 図 1 7 】



【 図 1 8 】



【 図 1 9 】



【 図 2 0 】

(A) (処理前)			
開始日(予定)	終了日(予定)	日数(予定)	日数(実績)
2011/6/24	2011/6/28	2d	3d
2011/6/26	2011/6/28	3d	
2011/6/29	2011/7/2	4d	

(B) (処理後)			
開始日(予定)	終了日(予定)	日数(予定)	日数(実績)
2011/6/24	2011/6/25	2d	3d
2011/6/27	2011/6/28	3d	
2011/6/30	2011/7/3	4d	

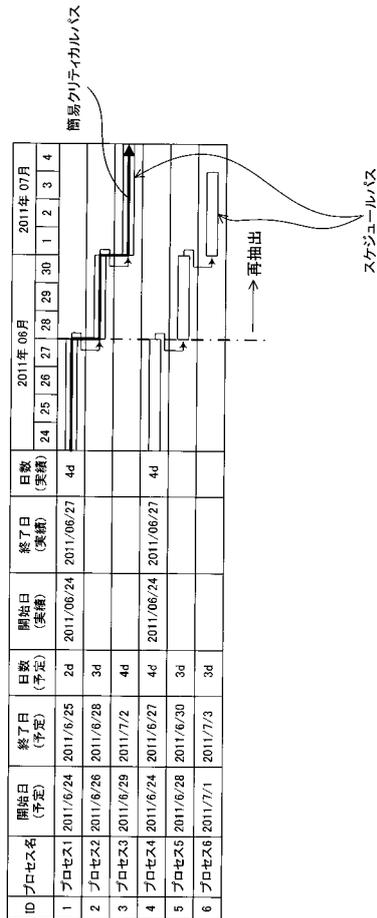
【 図 2 1 】

(A) (処理前)			
開始日(予定)	終了日(予定)	日数(予定)	日数(実績)
2011/6/24	2011/6/25	2d	
2011/6/26	2011/6/28	3d	
2011/6/29	2011/7/2	4d	

(B) (処理後)			
開始日(予定)	終了日(予定)	日数(予定)	日数(実績)
2011/6/24	2011/6/24	2d	
2011/6/25	2011/6/27	3d	
2011/6/28	2011/7/1	4d	

【 図 2 2 】



---

フロントページの続き

(72)発明者 後藤 一郎

神奈川県川崎市中原区上小田中4丁目1番1号 富士通アドバンステクノロジー株式会社内

Fターム(参考) 5B376 BA02