会計情報システムの文書化

武　田　嘉　孝

概要

本稿では、つぎのことを検討する。①ＡＩＳ（会計情報システム）を文書化することはなぜ必要か。②ＡＩＳにおけるデータの流れは、簡単な文書流れ図を作成することによって容易に説明することができるのか。③システム流れ図やデータ流れ図表は簡単に作成することができるのか。④プログラム流れ図、意思決定表、そして類似のツールがＡＩＳを文書化するために、どのように用いられているのか。⑤エンドユーザの文書化の必要性は何か。このような点を理解できるように論述をする。

1. はじめに

「不十分な文書化は、コンピュータシステムの有効な利用にとって、最大の障害の一つである」とも言われるほど、文書化は、会計情報システムにとって重要なものであると考えられる。たとえば文書化は、データを記録する手続き、コンピュータプログラムを実行するコマンド、ＡＩＳが実行される処理段階、システムを通じての論理的で実際的な流れを記述する。そして、会計士がなぜ文書化を理解する必要があるのかを考察し、文書化を行うためのツールを紹介する。一つの文書化ツールは、文書流れ図すなわち報告書の実際の流れを記述する図表、ＡＩＳによる請求伝票、類似のハード・コピー文書である。これらの流れ図はコンパクトな方式で、データ経路を絵によって表すために、文章的な記述よりも頁数の節約になる。ここでは先ず、文書流れ図を論じる。

二種類の文書化方法は、システム流れ図とデータ流れ図表である。これらはＡＩＳにおけるデータの論理的（電子的）な流れを記述するものである。システム流れ図とデータ流れ図表の両方共がしばしばデータ辞書、すなわちこれらの図表におけるデータの特徴を示すものを用いる。会計士というものは、特殊な会計適用業務のコンピュータ処理を理解するためのコンピュータ・プログラマである必要はないが、こうした処理がどのように行われているかについて、よく知っていることが重要である。その例としては、実行手順、プログラム流れ図、意思決定表、擬似コードを含む。最後の節で、そのような文書化ツールについても考察する。

2．なぜ文書化が必要か

文書化は、ＡＩＳの入力、処理、出力を記述する流れ図の全て、注釈、そして他の作成された通信を含む。文書化はまた、コンピュータシステム内でのデータの論理的な流れ、および従業員が適用業務の仕事を達成するために従わなければならない手続きを記述する。このように、そのシステムを記述するものが文書化である。つぎの５つの仕事は、ＡＩＳにとって、なぜ文書化が重要であるかを説明するものである。

①システムがどのように運用しているかを描くこと。

実行中のＡＩＳを観察するということは、たとえそのシステムが完全に手動のものであるとしても、それを知るには大変な調査と聞き取りが必要となる。たとえばデータの全ての流れや実行される関連した作業を説明するには容易なことではない。コンピュータ化されたシステムでは、システム内がブラックボックス化されているため、理解が困難である。それは処理が電子化され、そのために目で見えないことによるためである。他方、システムの入力、処理段階、出力について図表によって作成されていれば、業務が分かりやすく、全容を理解する助けとなる。これが、文書化の一つの目的、すなわちＡＩＳがどのように作用しているかを説明する手助けをすることである。またそうした文書化は、従業員はシステムの中でどのような作業をしているのか、会計士がそのシステムを改善するための手助けをしたり、管理者が統制するのを助けたりする効用がある。

②ユーザを教育すること。

文書化はまた、人々にＡＩＳがどのように作用しているかを学ばせる手助けとなるユーザ用のマニュアル、操作手続き方法、並びに同様の操作指導を含む。従業員は一般に、適用業務ソフトウェアに付随する特有なユーザ手引書を読むことを好まない。これらの指導的な資料は、参考補助として無価値なものとなっている。しかしながらこれらの文書化項目は、直接的か間接的に、ＡＩＳのハードウェアとソフトウェアの運用状況や、新しい運用上の問題を解決し、作業を改善する方法をユーザに教育するための手助けとなる。

③新しいシステムを設計すること。

文書化は、システム設計者が新しいシステムを開発し既存のシステムを保守するのに役立つ。他のシステム設計方法論に沿ったうまく作成された文書化は、コンピュータシステムにおける「緊急エラー」を訂正するのに費やされる時間を減少させるとともに、システムの失敗を減少させるために主要な役割を演じると言われている。

④他のシステムとの通信を標準化すること。

注釈記述は、それを作成する人次第でかなり多様化する。同様に注釈を読む個人が書かれた意図と違った意味で解釈するかもしれない。しかし共通の記号を用いるシステム流れ図やデータ流れ図表は、それを見る人々によって同じように解釈されそうである。このよ

うに文書化ツールは重要である。それというのもそれらは、既存のあるいは提案したシステムの簡明な描写をしたり、ユーザにこれらのシステムについて互いに通信するために役立てられるからである。

⑤ＡＩＳを監査すること。

最後に文書化は、監査証跡を描く手助けをする。たとえば組織体を監査するときに、監査人は先ず、内部統制を評価するであろう。そのような状況において、文書化は統制の長所と短所、監査の範囲と複雑性を判断するのに役立つ。同様に監査人は、標本の出力がそれらを発生させた原始取引まで追跡したがるであろう。（たとえば棚卸資産を原始仕入まで遡って追跡するような事）このようにシステム文書化は、監査人がこの追跡を実行するときの手助けとなる。

3. 文書流れ図

組織体内における文書の流れを絵によって表すための一つの有用なツールは、「文書流れ図」である。文書流れ図は、最終処理までに源を発する部門やサブシステムからの手操作のデータを追跡する。わずかな例を検討することによって、文書流れ図を理解することはおそらく最も容易なことである。

（1） 簡単な例

文書の流れ図を構築することは、特殊なシステムの文書を取り扱うさまざまな部門やグループを確認したり、図１で記号を用いて文書の流れを描くことによって始まる。たとえば、事務用消耗品を社内の中央補給品部門から支給を受けるときの事務作業を文書にすることを望んだとしよう。自部門の管理補助者は、つぎのようにその過程を説明するだろう。

①事例１

「消耗品を再注文するには、請求要請が必要です。たとえば文房具が必要なときには、２部の商品請求用紙に必要事項を記入します。私は中央補給品部門へ最初の１枚を送り、事務所のここに２枚目の用紙をファイルします。」

この例には二つの部門がある。一つは請求部門であり、もう一つは中央補給品部門であるそして文書流れ図上の（図２）見出しに、これらの部門の名前を付けることから始める。そのときに自部門の見出しの下に２部の商品請求用紙を描くであろう。その理由は、この部門が商品請求用紙を起こしているからである。２部の複写を示すために、１と２の番号を付けるであろう。

最後に、それぞれの文書がどこに行くかを示すであろう。すなわち１枚目は中央補給品

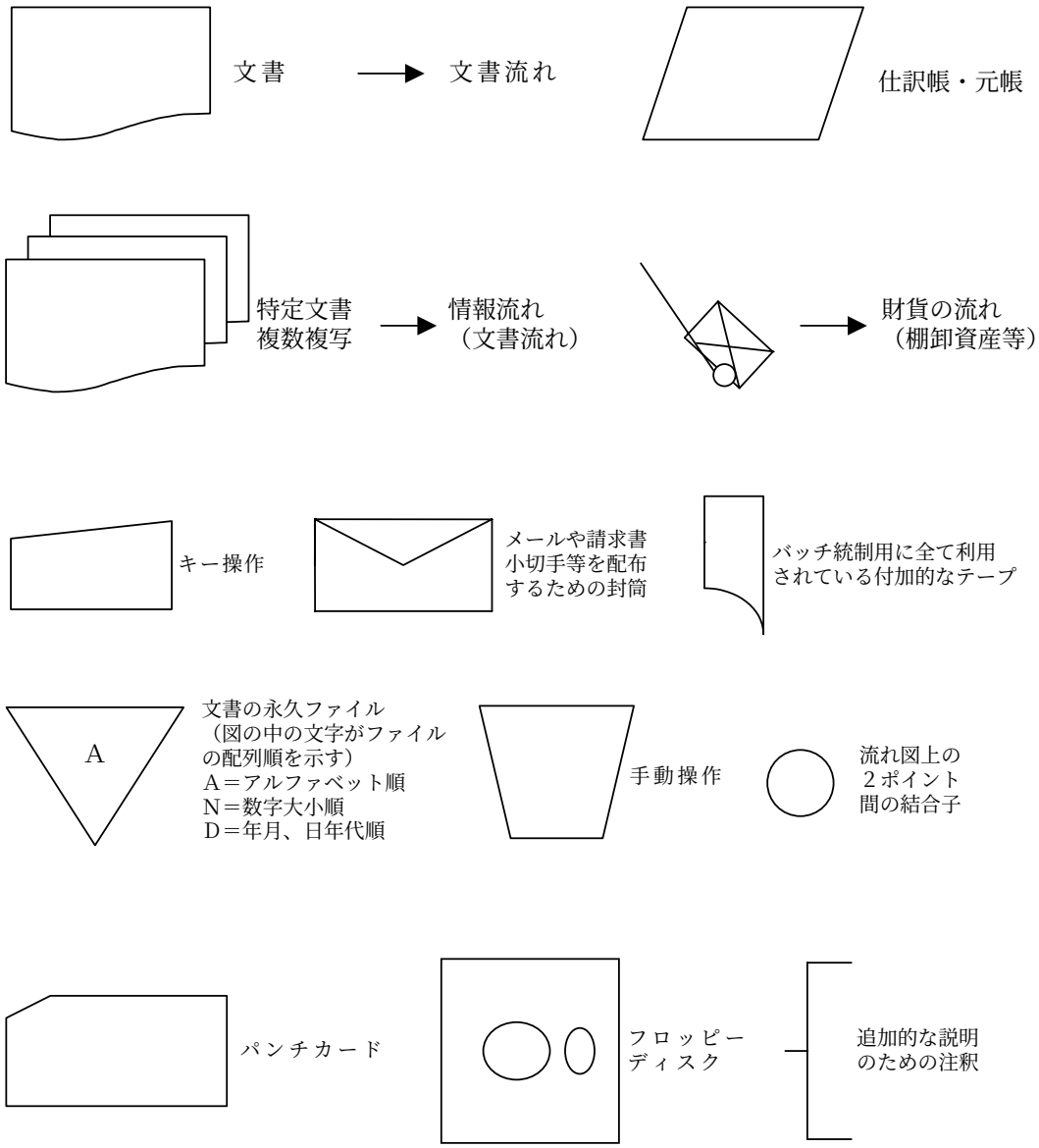


図１　一般的な流れ図記号

部門に、２枚目は自部門でのファイルとして保存される。文書が最初に出回るのは、それを記載した部門内である。実線は、ある場所から別の場所への実際の伝送を示す。その伝送された文書、それからそれを受領する部門に「到着」を示すために再び描かれるべきである。これらは図２で示すように描かれ、これでこの注釈に対する流れ図が完了する。

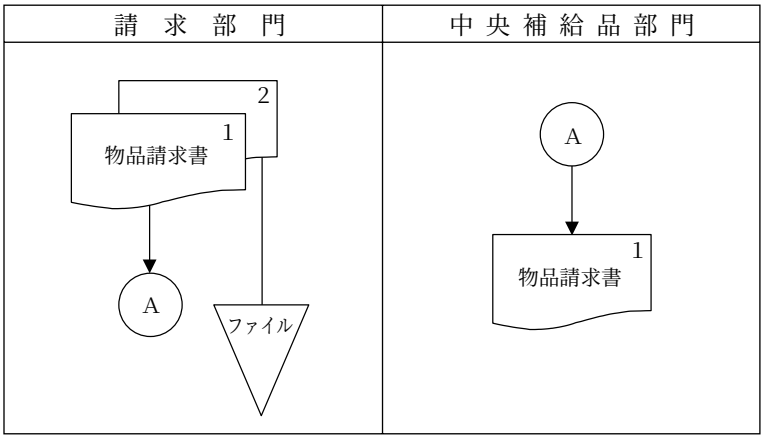


図２　簡単な文書流れ図

（２）複雑な例

文書流れ図を描くための方法をもう少し詳しく説明するために、Gary Company 社で新従業員を雇用する業務を取り上げよう。ある部門が欠員を出したときにそのプロセスは開始する。

②事例２

欠員を出した部門は、最初に職場欠員用紙に必要事項を記入し、それを人事部に転送する。それから人事部では、その職についての広告を出し、その補充を必要とする部門の支援を得て、応募者の面接を行う。

欠員が補充されたときに、人事部は職員雇用用紙の３枚綴り１組のものを作成する。人事部では、従業員社会保障番号によって整理された手動のファイルに最初の写しをファイルする。３枚目の写しは職場欠員用紙にホッチキスで綴じられ、その補充を必要とした部門に戻される。そしてその部門で、その用紙は従業員のラストネーム（ファミリーネーム）によってアルファベット順にファイルされる。

人事部は職員雇用用紙の２枚目の写しを給料支払部門に転送する。当部門はその用紙を新しい従業員のための給料支払記録をするための正式文書として使用する。そのように、その用紙に記入された情報は、給料支払オフィスに置いてあるオンライン端末を用いて当該部門のコンピュータシステムに直接にキー入力される。その職員雇用用紙は、それから参照用として、また従業員用紙が処理された証拠としてもファイルされる。

図３は、この例についての文書流れ図である。これを描くには、最初の段階は以前に示したのだが関係者を確認することである。このケースでは３者が存在する。イ．職務欠員をもった部門（図３では補充必要部門と呼ぶ）、ロ．人事部門、ハ．給料支払部門、である。

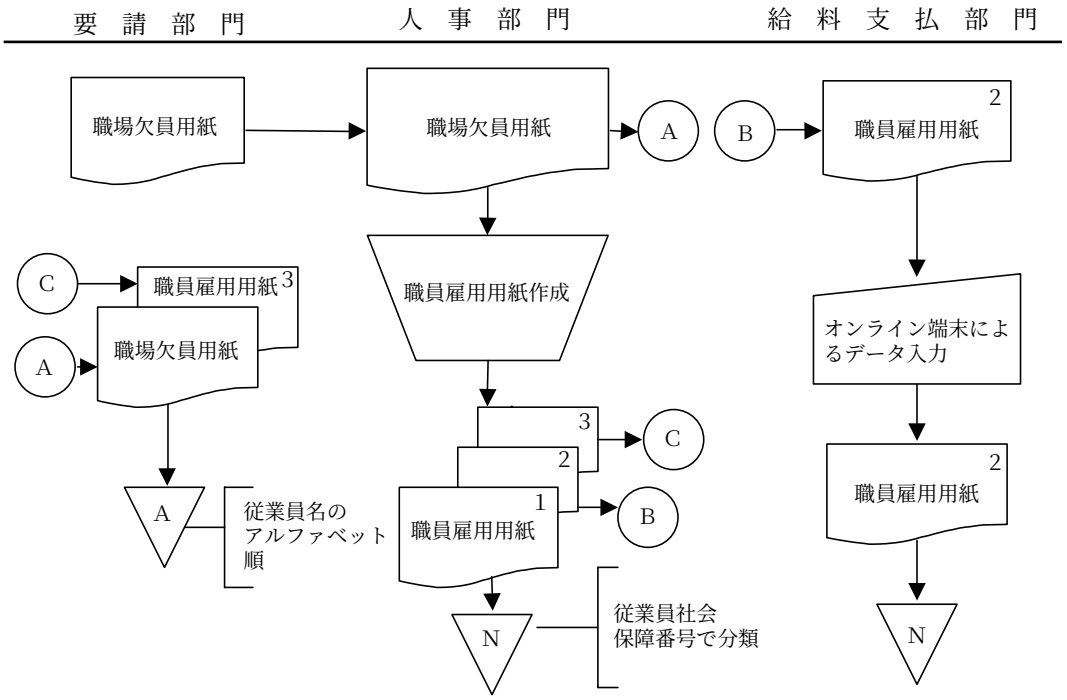


図３　新入社員を雇用する際に必要な文書の流れを示す文書流れ図

これらの部門それぞれは、文書流れ図の上に沿って確認される。

つぎの段階は、含まれる文書の確認である。そこには二つの主要なものがある。イ．職場欠員用紙は１通の写しとして作成される。ロ．職員雇用用紙は３枚１組みで作成される。実際に複数の写し用紙は通常、カラー番号化される。しかしながら文書流れ図において、これらは通常、単に番号化（付番化）され、カラー番号相当のものを説明するために、別々の頁が付けられる。

三番目の段階は、その文書が作成され、処理され、用いられる場所を示すことである。これは多分最も難しい仕事であって、文書流れ図設計者は、データの流れや処理活動の正確性を示すために、しばしばかなりの工夫を凝らさなければならない。図３は記述されている雇用手続についてのこれらの流れを図解している。沢山の文書伝送があるところでは、頁上の結合子が流れ線の複雑化を避けるために役立っている。それぞれの円に、文字のような特定の記号を与えて識別することが出来る。必要ならば欄外に結合子を用いることが出来る。図３は、A，B，C のような文字を用いて頁上にいくつかの結合子を用いることによって、流れ図を描くことの煩雑さを避け、完全な文書流れ図を示している。

（３）文書流れ図を作成するための指針

文書流れ図は、報告書や類似の文書における実際の流れを示すものである。それらを構

築するとき、分析家のいくらかはそれらの文書流れ図に物的な財貨の移動も含める。（たとえば、棚卸資産の受取り部門から保管場所までの移動などである。文書流れ図は典型的にこの仕事については、手押し車の記号を用いる。）文書流れ図は文書ではない情報の流れをも図示する。（たとえば、信用販売に至るまでに顧客の勘定残高をチェックするために販売員が電話すること。）このように、広義の文書用語は、あらゆる種類の組織上の通信や流れを含んでいる。他の種類の記号（たとえば、後で論じるシステムやプログラム流れ図記号）とは異なり、文書流れ図記号は標準化されていない。そして文書流れ図を描くことが、科学というよりも芸術的要素が大きいとしても、従わなければならない段階が存在する。図４は、文書流れ図を作るためのいくらかの重要な指針を示している。

文書流れ図はいかに重要であるか、ということについて会計士はそれらの有用性について同意してはいないのが一般的である。しかし大学等のシステムコースで学ぶ学生にとっては、最も基本的なスキルの一つである。会計情報システムを文書化しなければならないときは、文書流れ図を描くことが重要なスキルとなる。

つぎに示すのは、文書流れ図を作成するための重要な指針となるものである。

１．当システムに含まれる文書を作成したり、受取る全ての部門を確認せよ。
２．それらの関係する見出しの下に、それぞれの部門の文書や活動を注意深く直線にせよ。
３．会計文書のそれぞれの写しに、番号を付けて確認せよ。
４．文書のそれぞれの写しの配布について説明せよ。
５．多くの直線の付いた図表が互いに交差するのを避けるために、頁上あるいは欄外に結合子を使用せよ。
６．それぞれの組み合わせにおける、ａ「から」ａ「まで」といった）結合子の各一組は、独自の文字や数字を用いるべきである。
７．不明確な活動や記号を説明するために（もし必要であれば）注釈を用いよ。これらの注釈は、読む人に説明するために用いられる注記である。

図４　　文書流れ図を作成するための指針

4. システム流れ図とデータ流れ図表

文書流れ図は現実の文書に焦点を当てているのに反して、システム流れ図やデータ流れ図表は、ＡＩＳの処理や電子計算機のデータの流れに集中している。ここでは、システム流れ図やデータ流れ図表をさらに詳細に検討する。

（1）システム流れ図

システム流れ図はＡＩＳのデータの論理的な流れと処理段階を描く。図５は一般的なシ

ステム流れ図記号を図示する。これらの記号の大抵のものは、米国規格局（規格３．５）による標準化産業協定項目である。

いくつかのシステム流れ図は、一般的なものが存在しており、そしてそれは単にシステムの概観を提示するものであるに過ぎない。図６は高水準のシステム流れ図の一例である。システムの入力と出力は、一般的な入力・出力記号である平行四辺形によって明記される。もっと詳細なシステム流れ図では、入力・出力の特殊な形すなわち磁気ディスクや磁気テープ記号によって示されよう。

図６は、単一の処理すなわち給料支払を準備することだけに関連する。もっと詳細なシステム流れ図は、給料支払プログラムによって実行される全ての処理、及びそれぞれの処理の特殊な入力･出力を記述するであろう。（給料支払処理のための一つの例は、図８で示す。）そうした文書化の最も詳細な水準は、プログラム流れ図である。それはそれぞれのアプリケーションプログラムの論理を記述する。

（２）簡単な例

文書流れ図と同様に、簡単な実例を知ることによって、システム流れ図を理解することは、おそらく最も容易なことである。図７はつぎの事例についてのシステム流れ図である。

③事例３

Gabel Compapy 社は、毎月の郵送ラベルを作成するための雑誌購読者のファイルを維持する雑誌配達会社である。雑誌購読者は住所変更用紙や新規購読用紙を同社に直接に郵送し、そして同社の社員がオンライン端末を通じてシステムにその情報を入力する。オンラインコンピュータは一時的にこの情報を住所変更あるいは新規購読申込みのファイルとして保存する。このキー入力活動は継続的に実施される。そのことからこの作業を「日常処理」として特徴づけることができる。

１週間に１回、そのシステムは購読者マスタファイルを更新するために申込みレコードを用いる。このときに新しい購読者の名前と住所がそのファイルに追加される。そして住所変更した既存の購読者住所が更新される。マスタファイル保守処理報告書が、当該ファイルに行われた追加や更新を文書化するためのコンピュータプログラムによってまた作成される。

１ヶ月に１回、当社は雑誌の郵送のための郵送ラベルを作成する。購読者マスタファイルは、このコンピュータプログラムのための主要な入力ファイルとなる。二つの主要な出力は、ラベル自体とこれを実行したものを文書化する「郵便ラベル処理報告書」である。

図７でのシステム流れ図は、Gabel 社のコンピュータシステムによる会計データの流れを辿る。それはデータが発生する場所、データが一時的に格納される場所と処理された

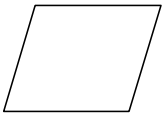
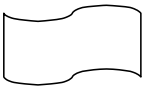
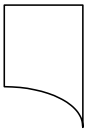
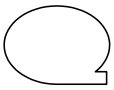
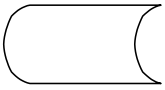
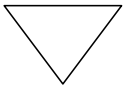
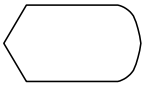
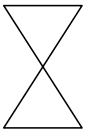
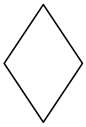
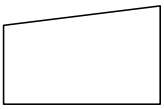
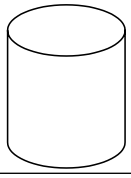
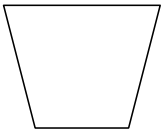
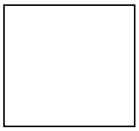
処理 主要な処理機能	入出力 全ての種類の媒体又はデータ 
パンチカード 控えを含むさまざまな パンチカード	穿孔紙テープ 
文書 全ての種類の紙文書と記録	転送処理テープ 試験済みあるいは付加的な機械化 されたテープまたは類似した一括 制御情報 
磁気テープ 	オンライン記憶装置 
オフライン保存 	表示 作図装置またはビデオ装置 で表示される情報 
接合 ２つ以上の他の集合から１つ以上の 集合の項目を作り上げること 	整列 分類・照合での操作 
手操作入力 オンライン装置を利用している コンピュータに与えたり与えら れたりする情報 	磁気ディスク 
手操作 機械の援助を必要としない 手動オフラインの操作 	補助操作 主な処理機能を補っている 機械操作 
入力する操作 キーボード装置を利用して いる操作	通信リンク 通信回線によるある場所から別の 場所への情報の電子的な伝送

図５　システム流れ図の作成記号（ＩＢＭ社の提供）

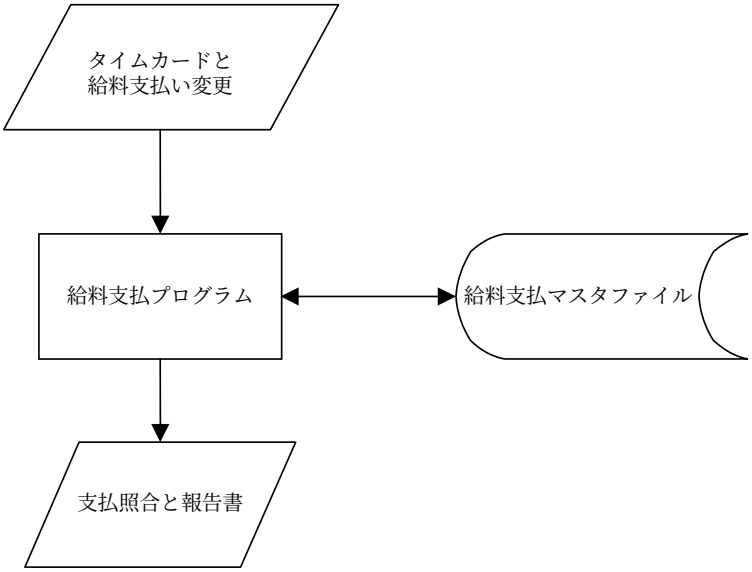


図６　給料支払処理のための高水準システム流れ図

データが現れた出力を確認する。図７におけるシステム流れ図は購読者の申込み用紙から始まり、キー入力段階からマスタファイル保全段階、最後に毎月の郵送段階までのデータの流れをこれらの書類に文書化する。

間接的に、システム流れ図はまた処理サイクル（毎日、毎週、毎月)、ハードウェア・ニーズ（ディスク・ドライバとプリンタ）と処理における潜在的な障害（例、手動入力）を示す。

図７で、システムの主要なファイル（変更要請記録の一時的なログファイルーデータ更新操作を記録したファイルーと購読者マスタファイル）とシステムの主要な制御報告を確認できる。

最後に、システム流れ図の各処理段階が通常一つ以上の制御報告を準備する必要がある点に注意しなければならない。これらの報告は、従業員がシステムによって発見されたエラーを訂正するための手助けとなる処理・制御情報を提供する。

④事例４

Gary Company 社は、毎週給料支払を準備する。基本的な情報は、ディスク上に保存された給料マスタファイル上に保全される。それぞれの給料支払レコードは、従業員ごとの（名前、社会保障番号、住所、控除額コード）基本的情報を含んでいる。時間給で働く従業員は、給料支払レコードに保存された時給支払率コード及び給料に関して給料支払レコードに保存された週給情報をもっている。

日常的にキー入力事務員は、人事部に配置された端末から直接的に給料支払ファイルに

変更を行う。ここに含まれるものとしては、昇給した従業員についての支払率の変更、あるいは転居した従業員についての住所変更などの例を挙げることが出来る。週末に人事部によって要請される変更は、一時的なディスクファイルから行われ、そして編集すなわち正確性と合理性のためのチェック(検査)も行われる。この編集作業を実行するコンピュータプログラムは、編集プログラムと呼ばれる。このプログラムからの出力は、それから給料支払マスタファイルトとともに、マスタファイル給料支払レコードを変更するために入力される。

給料支払マスタファイルへの修正が完了すれば、Gary Company 社は、給料支払を準備することができる。各週末には、会社内のさまざまな部門からのタイムカードが集められ、コンピュータセンターに持ち込まれる。はじめはタイムカードで得た情報は、単に磁気ディスクファイルに転送される。そしてそこで、システムが給料支払変更の編集に似た方法でそのデータを編集する。この後、実際の給料支払チェック（検査）が、別個の処理段階において準備されるはずである。実際の給料支払処理への入力は、給料支払マスタファイルと（編集された）タイムカード・データ・ファイルを含む。出力は（新年次支払と控除額を反映している）更新された給料支払マスタファイル、(コンピュータ制御総データと同様のものを示している）制御報告書、(支払った人、支払ったもの、チェック（検査）したものを示している）給料支払合計、並びに当然、給料支払チェック（検査）それ自体を含む。

（３）複雑な例

さらにもう少し複雑な問題を検討しよう。コンピュータ化されたデータ処理の標準的な例は、会社の給料支払処理である。給料支払処理システムは複雑なものであるが、それを簡素化したものが図８である。この図はつぎの情報に基づいている。

図８は、Gary Company 社における給料支払のコンピュータ処理のためのシステム流れ図である。そのシステム流れ図の左上側での情報の流れは、人事部門によって行われた給料支払変更要請から始まる。ディスクに直接その変更要請を転送するキー入力操作は、オンラインジョブ入力システムと呼ばれる。このキー操作、すなわち週ごとのこれらの変更要請の編集、並びに従業員マスタファイルを更新するためのこれらの変更要請の利用は、システム流れ図の左手部分を連続して下に読むように描かれている。

システム流れ図の右上側は、給料支払システムからタイムカード・データを追跡するものである。従業員タイムカード・データは、ディスクファイルにキー入力され、社会保障番号によっても分類される。最後に、流れ図の下の部分において、更新された給料支払マスタファイル及び従業員のタイムカード・データの編集されたディスクファイルは、週給支払の準備のために使用するために用意する。

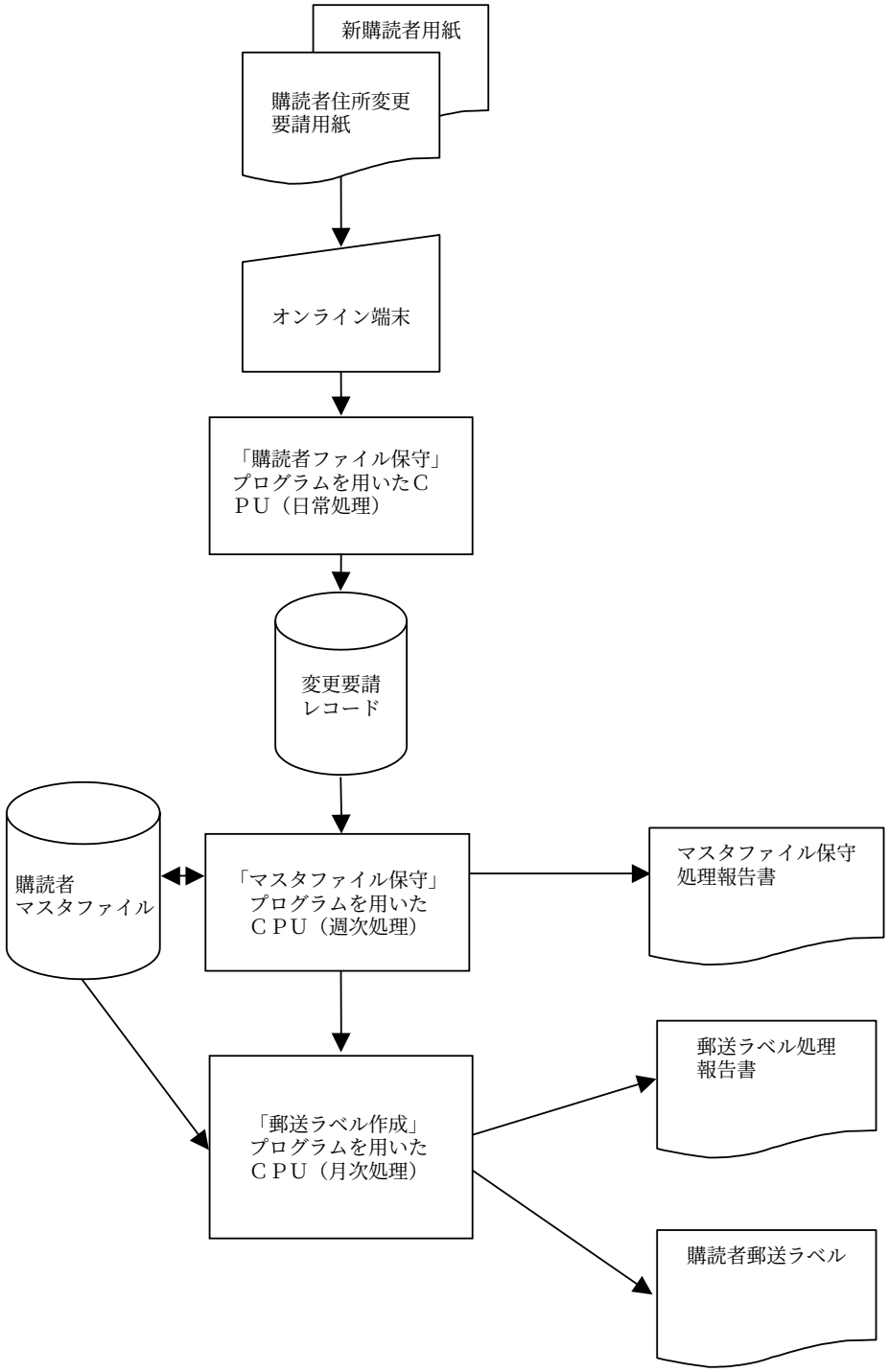


図７　購読者マスタファイルを保守し、月次郵送ラベルを作成するときに含まれるコンピュータ段階を例示しているシステム流れ図

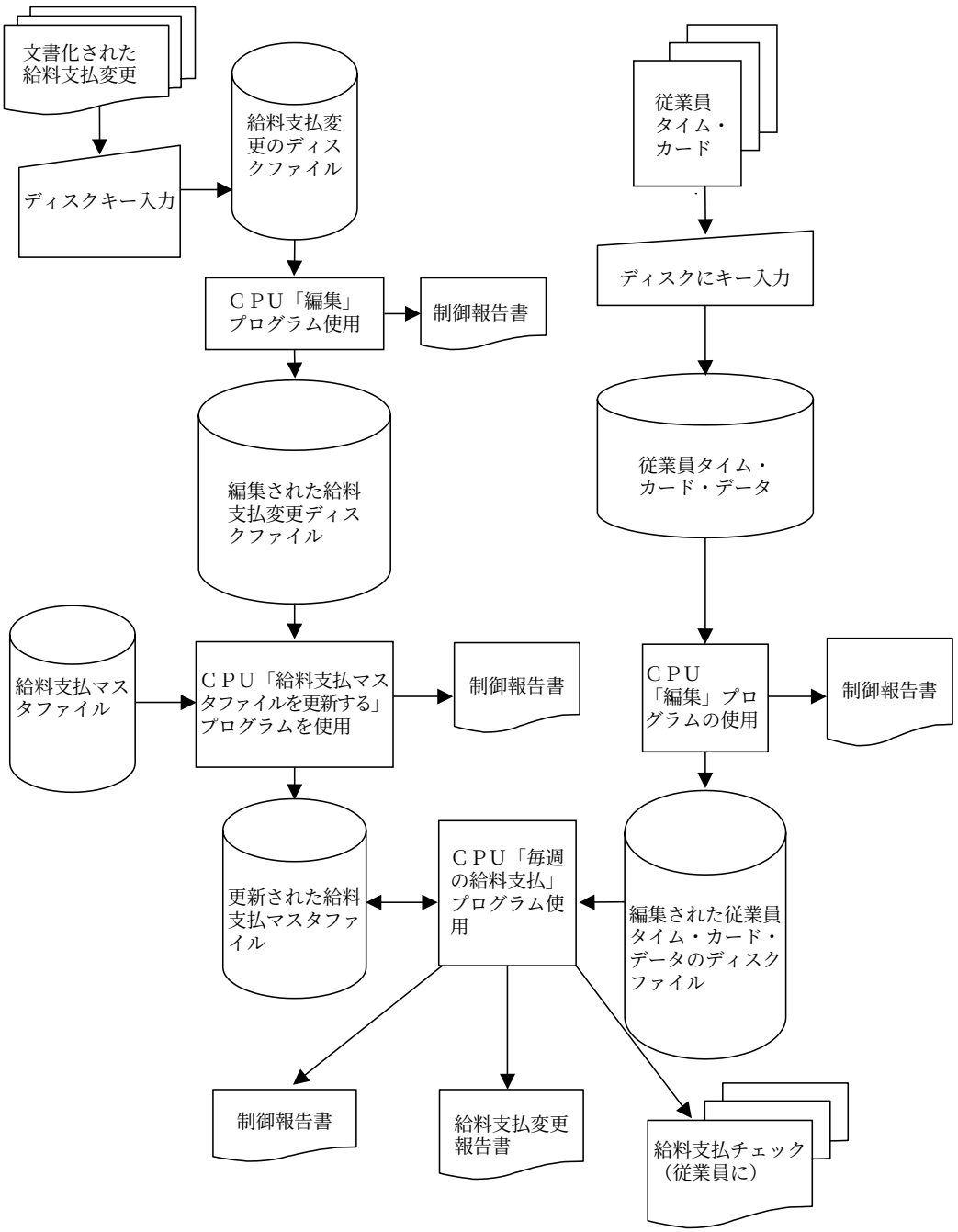


図８　Gary Company 社の給料支払のコンピュータ処理のためのシステム流れ図

（4）システム流れ図作成の指針

今まで示した例から分かるように、システム流れ図はＡＩＳのさまざまな処理段階を通じての電子ジョブ流れを描くことである。それゆえ監査証跡を追跡するものでもある、ということである。あるファイルのレコードが分類され更新されるときごとに、システム流れ図はこれを別々の処理段階において示さなければならない。概してこれは殆ど全てのＡＩＳにおいてある一時点における一つの段階を処理が進行する段階であり、それゆえシステム流れ図が処理段階を描写しなければならない方法である。

文書流れ図を描くときと同様、システム流れ図を描くことは、科学というよりもむしろ芸術に近いものである。システム流れ図を構築するための方法を性格に支配する厳密な基準が何もないけれども、図９はいくらかの一般的な指針を与えるものである。システム流れ図の有効性を認識するときに、アメリカ公認会計士協会や管理会計士協会の両方ともは、絶えずこれらの流れ図の実用的な知識を要求するテスト問題を専門試験に含めている。

1．流れ図は上から下、左から右へと読む。流れ図を描いたり読むときには、左上から始めるようにする。
2．適切な記号を用いよ。いくらかの流れ図表はプログラム流れ図での意思決定記号や、文書流れ図でのオフライン記憶装置記号のように予約されたものが存在する。
3．処理記号は必ず入・出力記号との間で見つけられる。これはサンドウィッチ規則と呼ばれる。
4．交差した線と混雑した流れ図を避けるために、結合子を用いなさい。
5．CASE 技法が、このジョブを容易にする最終の図面を設計する前に流れ図を下書きせよ。
6．明瞭化のために、流れ図に注釈記述やコメントを用いよ。これらはある処理に付けられた注釈記号あるいはファイル記号に含められたり、記号それ自体の中に含められるであろう。
7．あるファイルのレコードの連続性が重要であるならば、ファイル記号から水平に直線を描け。そしてその直線の下に文書を書きなさい。たとえば、レコードがアルファベット順に保存されているならば（A)、 数字順は（N)、年代順は（C）を示すようにせよ。もっと先の記述についての流れ図では、注が含められるであろう。

図９　システム流れ図を作成するための指針

（5）データ流れ図表

システム流れ図同様、データ流れ図表はＡＩＳを通じてのデータの流れの論理的な見解を提供するものである。それらは主としてシステム開発過程、たとえば既存のシステムを分析する技法、あるいは新システムを形成するための計画補助として用いられる。しかしながらデータ流れ図表というものの論理的な記述は、システムの物理的な特徴（たとえば入出力装置を記述すること）についての詳細なものは含めない。むしろデータ流れ図表は、

システムを通じての論理的な過程やデータの流れを描くものである。

（6）データ流れ図表記号

図10は、データ流れ図表において用いられる４つの基本的な記号を説明する。正方形は、顧客のようなデータ源もしくはデータの行き先を表す。このことを示すために、データ流れ図表は正方形内に顧客という言葉を入れる。データ流れ図表における円あるいは泡の記号は、データを変更もしくは変形する過程を示し、その円の中のラベルはこの変形を記述する動詞を入れる。最後の辺が空いた長方形は、データの記憶もしくは保存場所を表す。これは一般にある種のファイルである。最後に、データ流れは流れの方向を示す矢印によって、直線でデータ流れ図表上に示される。

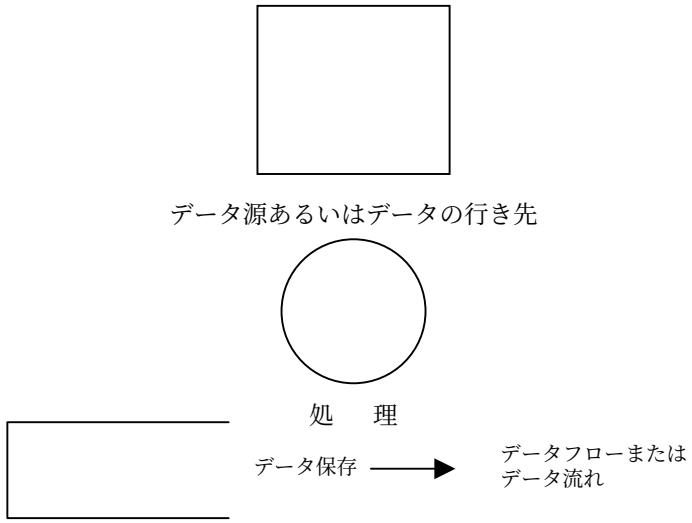


図 10　データ流れ図表のための記号

（7）分解

システム流れ図と同様、データ流れ図表は典型的に、かなりの詳細性のある水準で描かれる。高水準のデータ流れ図表は、アプリケーションやシステムの全体像を与えるために、先ず作成される。この一般的な概観は環境図表と呼ばれる。図11は、図６で記述したのと同じ給料支払処理のための環境図表の一例である。そのように図11は、アプリケーション外部のデータ源や行き先だけでなく、アプリケーションの入力と出力（給料支払処理）をも示す。しかしながらこの環境図表は、それほど詳細に示すものではない。

それが描かれた後で、それから環境変数は連続的なより低い水準の詳細さに分解され分析される。これらの後のデータ流れ図表は、もっと特殊なものを示す。たとえば、アプリ

ケーションの詳細な過程とそれぞれの処理段階とを結びつけた入・出力のようなものである。

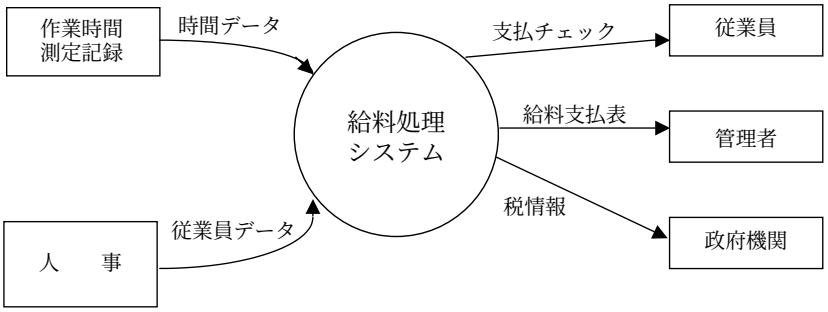


図11　給料支払い処理システムアプリケーションのための環境図表

たとえば図12は、図８で記述したのと同じ給料支払処理のためのデータ流れ図表を大雑把に示している。そしてそれは、図11のデータ流れ図表を分解したものである。たとえば図12は、タイムカードデータ処理、人事データ処理、給料支払いチェック（検査）処理、並びに給料支払報告書作成のそれぞれに対して、別個の記号を示している。その理由は、これらのものが別の処理タスクであるためである。

分解タスクは、この方式でまだ先のデータ流れ図表に継続する。たとえば図12の「給料支払報告書処理」という手続は、より低い水準のデータ流れ図表におけるいくらかの追加的な処理（たとえば、報告書を回収する処理、雑控除額報告書処理、支払表処理のようなもの）に分解されるであろう。この方法は、１組のデータ流れ図表が一つの階層の中に、一緒にリンクされるようになる。

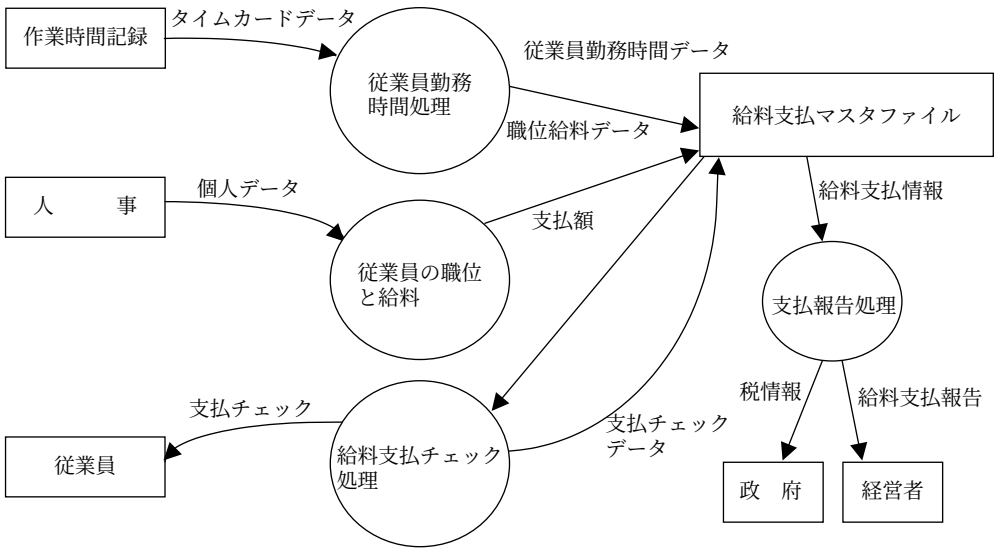


図12　給料支払処理システムのためのデータ流れ図表

（8）データ辞書

データ流れ図表は、データ要素とデータ構造の両方を含む。データ要素はデータ流れの最低のレベルであり、それゆえもうこれ以上小さな単位で分解することのできないものである。データ要素はデータ構造を形成するために連接されるものである。たとえば、「名前・住所」というデータ構造は、顧客の住所に顧客の名前をリンクするものである。しかしながら、顧客名と顧客住所はそれぞれのデータ構造である。

データ辞書は通常、これらの図表で用いられるデータ要素とデータ構造を記述するためにデータ流れ図表に伴う。データ辞書における入力は、データ構造の名前、データ構造の種類（データ要素、データ流れかデータ保存かのどちらか）、データ要素やそれらの関連によるデータ構造の内容、システム設計者コメントを含む。

複雑なシステムにとっては、データ辞書は作成したり維持されることが難しいはずである。この理由でシステム設計者は、しばしばコンピュータ化されたデータ辞書を作るために、CASE 技法のようなソフトウェアパッケージを使用する。つぎに CASE 技法をみることにする。

5. その他の文書化技法

ＡＩＳを文書化するためのツール（技法）については、フロー・チャート（流れ図）やデータ流れ図表の他に多くのツールがある。これらには、イ．プログラム流れ図、ロ．意思決定表、ハ．CASE ツール、ニ．擬似コード、がある。これらのツールは、専ら会計士よりもむしろコンピュータ専門家が用いるため、ここではそれらを簡単に紹介することにしよう。とはいえ会計士は、彼らコンピュータ専門家とは類似性をもっている。その理由は、たとえば情報システムを監査し、変更されたシステムについての設計を検査するときに、それらのツールを検討することを望むからである。さらにもっと大きな重要性を持つものは、ここでの五つ目のものとして追加する必要性のある、エンド･ユーザ･コンピューティングのための文書化技法である。

（1）プログラム流れ図

プログラム流れ図（図 13）は、コンピュータプログラムについての処理についての論理略図を描き、処理段階が実行される順序を示す。プログラム流れ図が一度設計されたならば、それは通常、認可を求めて管理者に示される。それからそのプログラム流れ図は、コンピュータプログラムそれ自体の命令を書くための「青写真」になる。そして勿論それは

プログラムそれ自体を文書化するのに役立てられる。

プログラム流れ図はシステム流れ図と同じ記号の多くを用いる(図５参照)。プログラム流れ図のための少数の特殊化された記号は、(処理論理における意思決定地点を示す) 菱形記号や（開始地点・停止地点を示す）楕円形の記号である。大抵の専門分析家は、コンピュータ図形のあるプログラム流れ図を作るために CASE ツールを現在用いている。手書きで記入するのであれば、標準的なプログラム流れ図記号のあるプラスチック製のテンプレートで済ますことができる。

システム流れ図やデータ流れ図表のようにプログラム流れ図は、さまざまな水準の詳細さで設計できる。最高水準のプログラム流れ図は、「マクロプログラム流れ図」と呼ばれ、データ処理論理の概観を与える。それ以下の水準のプログラム流れ図は、処理タスクを実行するために必要で詳細なプログラミング論理を示すであろう。図13は、販売報告アプリケーションのための詳細な（低水準）プログラム流れ図である。

（２）意思決定表

コンピュータプログラムが数多くの条件と連続的な行動過程を含むとき、そのプログラム流れ図は大きく複雑なものになりがちである。意思決定表は、それぞれの可能性を求めてどの適切な行動をとるべきかを示す条件と処理タスクとのマトリックス（行列）である。時々意思決定表は、プログラム流れ図の代替物として用いられる。もっと一般的には、流れ図の追加的なものとして用いられる。意思決定表を示すためには、事例５を検討してみよう。

⑤事例５

信用労働組合は年７％の利率で預金金利を支払う。５ドル未満の預金には利息の支払はしない。１年以上当労働組合に預金されている 1,000 ドル以上の金額には、通常の７％の支払に 0.5%の金利が上乗せされている。

図 14 は、信用労働組合がそれぞれの預金高にどれだけの金利を支払うかの決定をするのに役立つ意思決定表を示す。意思決定表は４つの部分から成っている。

イ．条件スタブ（モジュールすなわちプログラム単位）は、そのアプリケーションの潜在的な条件の概略を示す。

ロ．行動スタブは、採ることが可能な有効な行動の概略を示す。

ハ．条件エントリーズ（入力）は、起こりうる可能な組み合わせを描く。

ニ．行動エントリーズ（入力）は、それぞれの条件の組合わせに対して採られる行動の概略を示す。

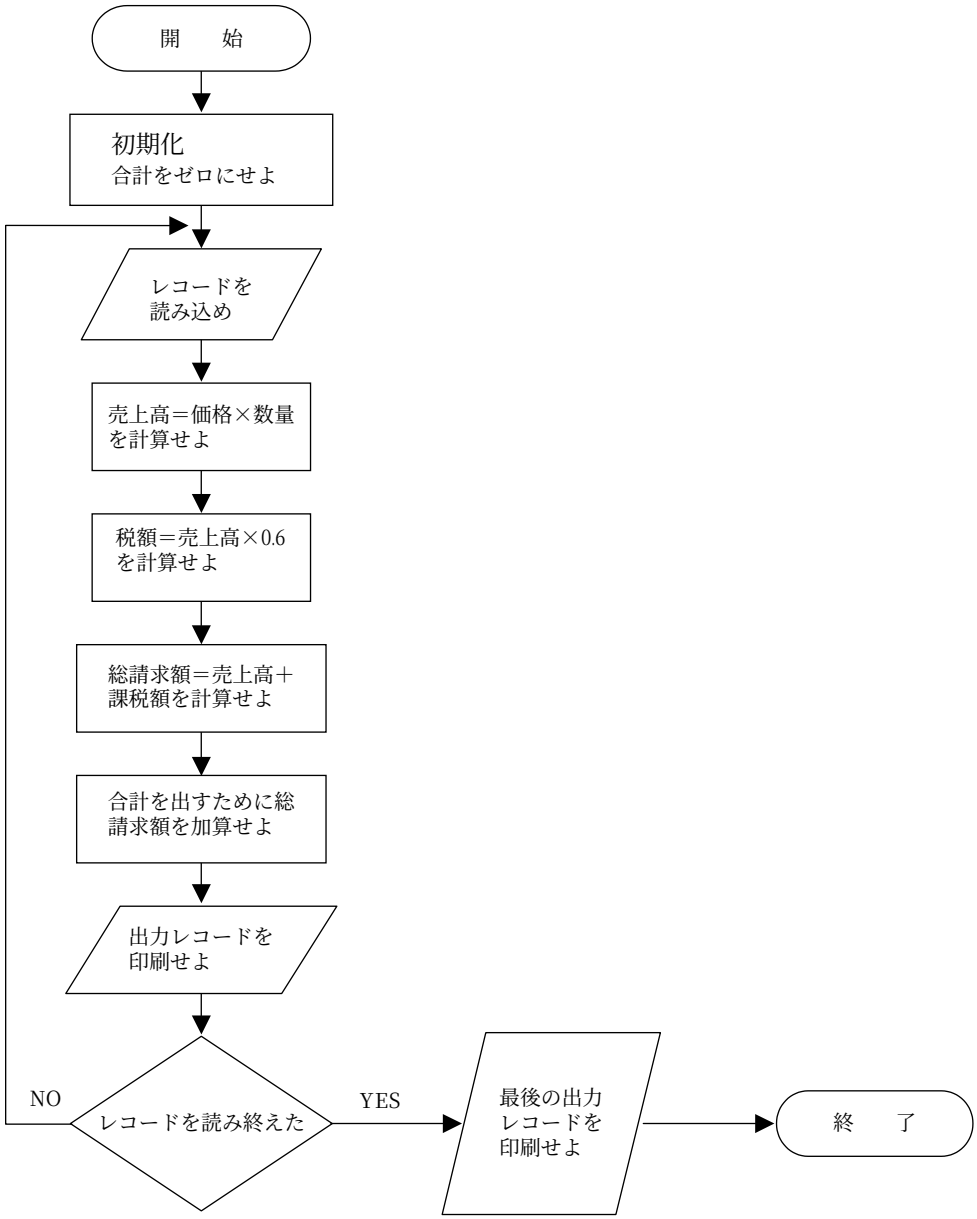


図 13　販売報告アプリケーションのための詳細なプログラム流れ図

		規		則		
		1	2	3	4	
	条件					
条件	預金残高５ドル未満	Y	N	N	N	条件
スタブ	預金残高 1,000 ドル未満	＊	Y	N	N	入力
	預け入れ期間１年以下	＊	＊	Y	N	
	行動					
行動	利息を払うな	X				行動
スタブ	７％の利息を支払え			X	X	入力
	7.5%の利息を支払え				X	

図 14　これは、信用労働組合が預金残高に対する利息を付けるために役立てられる意思決定表である。アスタリスク（＊）は、条件が行動に影響を与えないことを示す。

意思決定表の上に示した規則は、発生するであろう条件の組合わせやそれぞれにどのような行動を採るかを明らかにする。

具体的にそれを見ていこう。３つの条件がつぎのそれぞれの預金残高のデータ処理に影響を与える。①５ドル未満の預金、②1,000 ドル未満の預金、③１年以下の預け入れ期間の預金、である。これらの条件のそれぞれは明記したように、「Yes」か「No」でこの表から答えられる。図 14 は、手元の説明のための意思決定表である。そこで「Y」は yes のことであり、「N」は no のことである。その表のそれぞれの欄における Y と N の組合わせは、当該システムが出くわすそれぞれの可能な条件を説明するものである。X を用いて、意思決定表では又、それぞれの条件に対してどの行動方針を採るべきか（各預金残高にいくらの利息を支払うべきか）を示すものである。

意思決定表の主要な長所は、多数の条件に対する処理タスクをコンパクトに容易に理解しやすいように要約しているということにある。これによってプログラムの理解を深め、重要な処理可能性の脱落を少なく出来るようになる。意思決定表は又、新しい処理条件が生じるとき、あるいは組織の方針変更が既存の条件に対して新しい行動になるときに、有用な文書化支援として役立つ。この長所は特にＡＩＳにとって重要である。その理由は、組織の関心が財務データを処理するときの正確性や完全性に向けられているためである。

意思決定表の一つの短所は、プログラム流れ図のようにデータ条件がテストされ、処理行動が採られる順序を示すことが出来ないことである。これは主要な欠陥である。というのも会計データが検査され、処理される順序は、しばしば少なくともその検査や処理それ自体と同じくらい重要であるからである。第二の欠陥は、意思決定表は流れ図以上に文書化技法の理解を必要とすることである。最後に、意思決定表は作成するのに特別な作業を必要とする。そしてこの作業は、もしプログラム流れ図が作成されなければならないときには、有効な費用とは見なされないかもしれない。

（3）CASE ツール

CASE は、**C**omputer- **a**ssisted **s**oftware **e**ngineering の略称である。コンピュータ支援ソフトウェアエンジニアリングの頭文字をとり、ケースツールと呼ばれている。これは、流れ図を描いたり修正したり、グラフィックスや画面設計を作成したり、報告書書式を開発するよな文書化タスクを自動化するものである。このようにケースツールは、ワープロが文書をテキストにするものを流れ図にすることができる。

大抵のケース製品は、パソコン上で実行するものである。その例にはつぎのものがある。アプリケーション･ファクトリ（Contex Corporation 社）、ブリーフケース（South-Western Publishing Company 社）、デザイン／1（Andersen Consulting 社）、並びにフォアサイト（Athena System 社）などがある。ケースツールは、システム設計者がきわめて広範な出力の配列を作ることを可能にする。それらのものとしては、データ流れ図表、入力関連図表、システム流れ図、プログラム流れ図、レコード様式、プロセス・モデル、データ入力画面、報告書書式、メニュー画面構成図、顧客対応のユーザ・インタフェースなどが含まれている。大抵の CASE パッケージは又、データ辞書を作るためのジェネレータ（生成プログラム）及び文書化を行うためのワープロを含む。高度の CASE パッケージは、画面設計あるいは詳細なプログラム流れ図に基づいたコンピュータコードを規則正しく生成することが出来る。

CASE ツールは、ユーザが文書化を迅速且つ堅実に生成し、変更の要求があったときにもこの文書化を修正することを可能にする。たとえばアプリケーション・ファクトリを用いて、デュポン社（Dupont Corporation 社）は、ナイロンの引き伸ばし－包装製品について3万ドルのコストでコンピュータシステムを設計することが出来た。それも当初の見積額が 26 万 8 千ドルであった。

いくらかの CASE パッケージは又、システム設計努力における矛盾を発見したり、単一の入力から全ての変更をなすことが出来る。しかしこれらのパッケージは、作成したいと言われるものを作るだけである。それらは想像力や創造性を欠き、有効に用いるためのユーザの訓練が必要である。

（4）擬似コード

擬似コードは「模倣コード」を意味する。これはコンピュータが理解するプログラム言語で書かれていないために、本当のプログラムコードではない。その代わりに、擬似コードはプログラム論理を記述するための構造的な形式で英語の単語を使用する。こうした理由で、擬似コードの形式は時々「構造英語」として言及される。擬似コードは、プログラ

ムだけではなくシステムの論理を記述するためにも用いられる。

擬似コードで用いられる言葉は、add（加算）、subtract（控除）、do（～せよ）のような限定的な一組の動詞や、if（もし～ならば）、then（そのときに）、そして else（そうでなければ）のような条件用語である。これらの条件用語は意思決定表やプログラム流れ図で示されるロジック（論理）を記述するために用いられる。連続した関連性を示すために文章や用語を字下げすることについての規則を含む擬似コードについてのいくつかの申し合わせがある。

擬似コード、意思決定表、プログラム流れ図は全て、構造化プログラミングを促進するために用いられる。構造化プログラミングは、上から下へモジュール形式で注意深く組織化されているコンピュータプログラムを構築するために用いられる。構造化プログラミングは、システム分析家がそれを用いると、今までよりもっと迅速にプログラムを開発できるようになる。その理由は、プログラマの責任がモジュール単位で割り当てられており、又そのプログラムがモジュール単位で検査され、修正され、文書化されるためである。この構造化は又、会計情報システムにとっても重要である。その理由は、処理制御が構造化プログラムにおいて同一のものであることを証明することが可能となるからである。そして又、構造化プログラムのための文書化が、もっと容易に理解できる潜在能力を持っているからである。

（5）エンドユーザ・コンピューティングと文書化

エンドユーザ・コンピューティングは、コンピュータの専門でない従業員がコンピュータ適用業務に携われるようにするための能力に関連する。今日、我々はこの「コンピューティング」の殆どを自然に当然のものとして受け入れている。その例として、従業員がワープロ、表計算、税パッケージなどで巧みにデータ処理をしている。それというのもこうしたパッケージの多くは、エンドユーザがアプリケーションソフトを彼ら自身が利用するために、特別に開発されたものであるためである。

エンドユーザ・コンピューティングは広範な文書化を必要とする。それというのも、エンドユーザがコンピュータ・ソフトウェアを使用し、アプリケーション業務を実行できるようにするためのものであるため、具体的にして練習しやすいマニュアル、指導書、参考手引書を必要とするからである。既存のソフトウェアの新改訂版は、常に学習曲線の最低ラインにいて経験のない初心者用に向けて作られている。そしてそのようにして、いかに物事を達成するようにするか（あるいは我々の間違いを取り消せ）を再学習するために、主として文書化に依存させるようにする。

エンドユーザ文書化は又、つぎの時に重要である。エンドユーザが彼ら自身のシステムを生成する時である。たとえば、表計算モデルかデータベースシステムなどの自己開発に

よる文書は、それらを作成した従業員に、これらのアプリケーションを文書化するための責任を課すことである。不幸にもこの文書化作業は、しばしば見落とされ、きわめて実行することが少ないため、大目に見られることになってしまう。そのような手落ちは、かえって費用がかかることになる。たとえば、他の従業員が使用しなければならないとき、あるいはシステムを変更する必要が出てきたときに、これらの業務を達成するための基本的な文書化がなされていない場合、無駄な時間が浪費される。

したがって、いかなる特殊なエンドユーザ・アプリケーションであっても文書化すべきである。特殊な項目は、勿論アプリケーションによって異なるであろう。たとえば、ビジネスでは、ワープロ文書を文書化したり、他の人が後で見つけやすいように報告書の中にこれらのファイルをはめ込むために、体系的なファイル名を用いることは便利であることをしばしば知るであろう。同様に図15は、表計算アプリケーションを文書化するための基本的な考え方を若干提供するものである。

1. 開発者の名前
2. アプリケーションが格納されるファイルの名前
3. アプリケーションが格納されるディレクトリとサブディレクトリの名前
4. アプリケーションが最初に開発された日付
5. アプリケーションが最後に修正された日付とそれを修正した人の名前
6. アプリケーションが最後に実行された日付
7. 問題があった場合に電話をする人の名前と電話番号
8. 当該システムによって用いられる外部データ源
9. アプリケーションにおいて作られた重要な仮定
10. 仮定の変更、「何・もしも」の質問に答えるのに修正が必要とされる重要なパラメータ
11. アプリケーションで用いられるレンジ（範囲）名と表計算におけるロケーション（場所）

図15　表計算アプリケーションのためのエンドユーザ文書化

6. おわりに

これまで検討してきた内容を要約しよう。ＡＩＳを文書化することと結びついた５つの機能は、①システムがどのように作用しているかを説明すること、②他の人に教育するのに役立つこと、③他人が新しいシステムを設計することを支援すること、④他人との通信を促進すること、そして⑤監査人に情報を提供すること、であった。書かれた注釈はＡＩＳを文書化するために用いられるけれども、いつももっと能率的であるいくらかのグラフィックツールは有効なものである。そのような一つのツールは、文書化流れ図である。文書化流れ図は例えば、つぎの概観を提供することによって、ＡＩＳを通じての文書の実

際の流れを記述する。すなわち、文書はどこで作成されるのか、どの部門がそれを受け取り、どの部門でそれを検査するのか、彼らはどのような活動を起こし、これらの文書がどこで保存され又、廃棄されるか、ということである。

他の二つの種類の流れ図は、システム流れ図とデータ流れ図表である。システム流れ図は、ＡＩＳを通じてのデータの電子的な流れを記述し、どの処理段階とファイルが用いられまたそれはいつ用いられるのかを示す。そして全体のシステムの概観を提供する。データ流れ図表は又、入力、処理、出力の実際の装置あるいはタイミングに関してよりもデータの流れや変換に、より多く集中するシステムの論理的な考え方を提供する。データ流れ図表は、データ辞書と共に、システムを記述するための強力な文書化ツールを形成し、しばしばシステム開発作業に用いられる。

他の五つの文書化ツールが論じられたが、それらは、プログラム流れ図、意思決定表、CASE ツール、擬似コード、エンドユーザ・コンピューティングと文書化、である。会計士は会計情報システムを評価したり設計するためのプログラマである必要はない。しかし彼らは、一般的な言葉でこれらのツールがどのように作用しているか、特にエンドユーザ・コンピューティングと文書化を理解すべきである。

引用文献

Moscove, Stephen A, 1997, *Core concepts of accounting information systems*, John Wiley & Sons, Inc., pp. 48-74.