

61—S—001

システム監査支援ツールの 調査研究報告書



昭和 62 年 3 月

JIPDEC

財団法人 日本情報処理開発協会

この報告書は、日本自転車振興会から競輪収益の一部である機械工業振興資金の補助を受けて昭和61年度に実施した「システム監査支援ツールの調査研究」の成果をとりまとめたものであります。

序

高度情報化社会の進展につれ、社会におけるコンピュータへの依存度は益々強く、情報処理システムの質に対する要求は一段と厳しくなりつつある。

一方、システムの一層の巨大化、複雑化と遠隔地からの通信回線を通してのオンライン化の普及にともない、情報処理システムの脆弱性が増大し、システム監査の重要性が更に強調されるようになった。

システム監査の必要性は、既に1960年代初期の頃より提唱されて来たが、我国では特にその実施がおくれ、現在、米国では約85%の企業がシステム監査を実施しているのに対し、我国は約10%強という状況である。

今日、情報処理システムの監査を効果的に行わんとした時の障害は、先ずそのブラックボックス性と情報量の多さであり、これらを克服するためには、コンピュータ自身を用いたシステム監査支援ツールが必要不可欠となって来ている。

従来から、米国を中心に、種々のシステム監査支援ツールの開発が試みられ、またその実用実績もかなり存在する。しかしながら、急速に変化する情報処理技術への対応や、情報処理形態の多様化への対応は十分とは言えず、また、コンピュータ・ソフトウェアの開発に関わる技術的諸問題への監査側の対応もまだ模索的な段階に止まっている。

以上のような背景をふまえ、本プロジェクトでは、特に情報処理システムの信頼性と安全性に重点をおき、その監査のための新しい支援ツールの検討を行うものである。

2年計画の第1年度目にあたる本年度は、従来までのシステム監査支援ツールの実態と内外の使用実績、および現状での問題点等を総合的に把握し、新しいツ

ールとして、どのような機能が求められるかを検討した。

なお、本プロジェクトの遂行にあたり、我国におけるシステム監査の權威であられる下記の方々に、システム監査支援ツール研究委員会メンバーとしてご参画いただき、極めて有意義な討論およびご意見をうかがうことが出来た。ここに深く感謝を表する次第である。

(五十音順)

研究委員会	委員長	石 崎 純 夫 氏	(産業能率短期大学)
	委 員	浅 輪 寿 男 氏	(住友スリーエム(株))
	"	上 園 忠 弘 氏	(日本アイ・ビー・エム(株))
	"	大 井 正 浩 氏	(株三菱銀行)
	"	高 橋 富 夫 氏	(株富士通)
	"	堀 内 一 氏	(株日立製作所)
	"	松 尾 明 氏	(監)中央会計事務所)

昭和 6 2 年 3 月

目 次

序

1. 総 論	3
1.1 目的と背景	3
1.2 調査研究の概要	4
1.2.1 システム監査支援ツール・技法	4
1.2.2 システム監査支援ツールの現状調査	5
1.2.3 情報システム開発技法とシステム監査	12
1.2.4 システム監査支援ツールの課題	13
2. システム監査支援ツール・技法	19
2.1 SACレポートによるシステム監査支援ツール・技法の概要	19
2.2 システム監査の計画と管理を行うためのツール・技法	19
2.3 アプリケーション・プログラムの正確性をテストするためのツール・技法	22
2.4 監査対象のトランザクションの選択と監視を行うためのツール・技法	25
2.5 データの正確性を検証するためのツール・技法	28
2.6 プログラムの分析をするためのツール・技法	29
2.7 コンピュータ・センターの運用の統制を検証するためのツール・技法	32
2.8 アプリケーション・システムの開発を監査するためのツール・技法	33
2.9 システム監査支援ツール・技法の比較	36
3. システム監査支援ツールの現状調査	41
3.1 米国における現状	41
3.1.1 調査の概要	41
3.1.2 A製材会社	45
3.1.3 B銀行	56
3.1.4 C製薬会社	64
3.2 国内における現状	72
3.2.1 調査の概要	72
3.2.2 調査結果の評価・分析	78

4. 情報システム開発技法とシステム監査	95
4.1 情報システム開発の動向	95
4.1.1 情報システムの発展段階とシステム化課題の変遷	95
4.1.2 拡大期における方法論	96
4.1.3 拡大期の方法論がもたらしたもの	98
4.2 拡大期方法論の欠陥と統制期における方法論	99
4.2.1 拡大期末期の情報システム・イメージ	99
4.2.2 拡大期方法論の欠陥	100
4.3 内部統制を前提としたシステム開発方法論	102
4.3.1 システム開発、運用における統制のあり方	102
4.3.2 システム開発、運用における統制と監査のフレームワーク	104
4.4 システム・ライフサイクルとシステム監査	107
4.4.1 システム・ライフサイクルとその統制	108
4.4.2 開発、運用を一貫して統制できる機構	109
4.4.3 開発統制としてのシステム・ライフサイクル管理	113
4.4.4 保守統制としてのシステム・ライフサイクル管理	115
4.5 開発、保守の統制技法	116
4.5.1 開発統制のための技法	116
4.5.2 保守統制のための技法	119
5. システム監査支援ツール・技法の試用実験	125
5.1 実施目的	125
5.2 監査対象の概要	125
5.3 監査計画	130
5.4 監査の実施	138
5.5 監査報告	144
5.6 評価考察	144
6. システム監査支援ツールの課題	153
6.1 現在のシステム監査支援ツールの課題	153
6.1.1 機能上の課題	153
6.1.2 監査実施上の課題	157

6.2	新しいシステム監査支援ツールの考え方	162
6.2.1	情報処理システムの変化に対応するツール	162
6.2.2	ソフトウェア作成段階における監査ツール	169
6.2.3	システム監査業務の作業環境の改善	172
結 言		183
付 録	システム監査支援ツール・技法の実態調査票	187

1 . 総 論

1. 総 論

1.1 目的と背景

近年高度情報化の急速な進展につれて、情報化の陰の部分とも言える問題が顕在化して来ている。その代表的なものの1つがセキュリティ問題であり、それに対処するものとしてシステム監査の重要性が強調されつつある。

「システム監査とは、監査対象から独立した客観的な立場で、コンピュータを中心とする情報処理システムを総合的に点検・評価し、関係者に助言・勧告することをいい、その有効利用の促進と弊害の除去とを同時に追求し、システムの健全化をはかるものである。」

と定義されているように、システムの信頼性、安全性、効率性の監査がその3つの柱となっている。

本プロジェクトでは、セキュリティとの関連から、このうち特に信頼性と安全性の監査に焦点をあて、さらにそのためのシステム監査支援ツールの検討を目的としている。最近の情報処理システムの高度化・複雑化にともない、システム全体におけるソフトウェアのウエイトが増大し、その結果、システムのブラック・ボックス性がさらに増大して来ている。

従来のシステム監査は、ともすれば、このような実態から来る監査実施上の困難さを避け、概して表層的な範囲の監査しか実現し得なかった。そのような欠点を改善し、システム監査の本来の目的達成のためには、可能な限り、システム内部が透明であることが望ましく、ソフトウェアの不可視性を少しでも多く排除し得るより強力な支援ツールの存在が必要不可欠となった。

本プロジェクトでは、このような背景をふまえ、従来までのシステム監査支援ツールの実態と問題点とを海外事情をも含めて幅広く調査し、新たに求められる機能とその実現の可能性につき検討することを目的とする。なお、本プロジェクトは、昭和61年度、62年度の2年間にわたり実施することとしており、61年度は、その1年度目として、下記の内容につき検討をおこなった。

- (1) 現在のシステム監査支援ツール・技法の種類と目的別分離
- (2) システム監査支援ツールの内外における利用上の現状調査
- (3) 情報処理システムの開発技法とシステム監査との関わりについての検討
- (4) 現在のシステム監査支援ツールの課題と新しいツール機能に関する検討

1.2 調査研究の概要

1.2.1 システム監査支援ツール・技法

システム監査の技法や支援ツールと呼ばれるものは、従来から既に多数紹介されている。

特に1977年にIBMとSRIが協力して調査し、まとめたSACレポートは有名であり、その後のシステム監査技法や支援ツールのバイブル的存在となっている。

1977年のこのレポートが今だにその力を失わないのは、1つにはその後の研究開発が必ずしも活発には行われて来なかった故とも言えよう。

最近SACレポートの分類方法に若干の異論をとる向きもあるが、より適切な定評ある他の分類法も存在しないことから、本報告等に於てもSACレポートにならい、現在の28種類の支援ツールおよび技法を以下の7つに分類し、それぞれの目的、機能、要件、形態の説明を行った。

- ① システム監査の計画を立てたり、作業の管理を行う時に用いた方法
- ② アプリケーション・プログラムの正確性をテストする時に用いた方法
- ③ 監査対象のトランザクションの選択と監視を行う時に用いた方法
- ④ データの正確性を検証する時に用いた方法
- ⑤ プログラムの分析をする時に用いた方法
- ⑥ コンピュータ・センターの運用の統制を検証する時に用いた方法
- ⑦ アプリケーション・システムの開発を監査する時に用いた方法

一方これらの技法および支援ツールを、以下の5つの離点から見直し比較したのが表2-1(37ページ)である。

- (1) 監査対象がデータ・ファイルか、アプリケーション・プログラムか、コンピュータ・センターか、OSか
- (2) 企画・開発・運用などシステム・ライフサイクルのどの段階の監査に適用するか
- (3) 準拠性・信頼性・安全性・効率性などを始め、監査目的は何か
- (4) 監査データの収集、分析、検証、プログラムの正確性やロジックの検査など、どのような機能があるか
- (5) 汎用性、専用性、組込み、分離などツールのあり方

この表によれば、本プロジェクトの主たる目的であるシステムの信頼性および安全性の監査のための代表的支援ツールとしては、以下のようなものが

あげられる。

- ① テストデータ法
- ② 総合テスト法
- ③ 平行シミュレーション法
- ④ 汎用監査ソフトウェア
- ⑤ 端末監査ソフトウェア
- ⑥ スナップショット法
- ⑦ トレーシング法
- ⑧ マッピング法
- ⑨ コード比較法

1.2.2 システム監査支援ツールの現状調査

現状調査としては国内と海外（米国）の両方を行った。

(1) 国内における現状

米国に比べ我国のシステム監査の普及は大幅に遅れており、米国では現在、全企業の85%がシステム監査を実施しているのに対し、我国では10%強でしかない。

特にシステム監査支援ツールを十分駆使して監査を行っている組織はさらに限られるのが現状である。

そこで本調査は、金融機関を始め、我国でのシステム監査の先進企業と思われる31社を選び、アンケートを行った。アンケート内容は付録に示す。

本アンケートでは31社中22社（回答率71%）から回答を得ることができた。またツールに関しては、1社で複数のツールを使用しているケースもあるため、合計30票の回答が得られた。

以下にアンケート結果の概要を示す。

① ツール利用に関する結果

A票（使用中のツールに関する質問）の回答を集約したのが表3-4（80ページ）である。次のような傾向が特徴的である。

イ. 使用している支援ツールまたは技法のうち、最も多いものは、汎用監査ソフトウェア（11件）であり、次いでテストデータ法（6件）、総合テスト法（4件）、コード比較法（3件）、平行シミュレーション

ン法（２件）などがある。

ロ、汎用監査ソフトウェアとコード比較は市販プロダクトを用いているが、他は自作が多い。

ハ、EASYTRIEVE、FOCUS等の第４世代言語を用い、システム監査支援ツールの代用としている企業が４社ある。

ニ、被監査システムがオンライン系も可能なものが２７件中１５件あり、バッチ系１２件より多い。

ホ、監査対象としては、汎用監査ソフトウェア系のものは殆んどデータ・ファイル対象であり、テストデータ法系統のものは、殆んどアプリケーション・プログラム対象である。

ヘ、使用段階は運用段階（２７件）が圧倒的であり、他は開発段階（１１件）、企画段階（１１件）となっている。

ト、監査目的は、信頼性（２７件）、正確性（２４件）、安全性（２２件）、効率性（１１件）、有効性（１０件）、採算性（９件）と分散しており、殆んど支援ツールが、複数の目的に併用されている。

チ、使用効果に対しては、殆んど支援ツールが大なり小なり、監査作業の効率、監査結果の精度、監査範囲の拡大、監査期間の短縮などいづれにも貢献している。

② ツールを使用していない企業の意見

イ、ツールを使用していない理由を回答した企業が１１社あるが、その殆んどは、システム監査の体制がまだ支援ツールを使用するような段階に至っていないとしている。現在の支援ツールが使いにくいという意見は１件だけであった。

ロ、使用して見たいツールとしては、テストデータ法（４件）、汎用監査ソフトウェア（４件）を筆頭に、２０数件の提示例の殆んどに分散している。

ハ、監査対象や監査段階の予定は、支援ツールを既に使用している企業の回答と殆んど差がない。

③ システムの信頼性、安全性監査のためのツールに対する意見や要望
４０件もの多様な意見が寄せられた。

現在の支援ツール、あるいは開発用の支援ツールの流用、ユーティリティの流用などを含め、使い方によっては、現状でもかなり有効なもの

が存在するという意見もあるが、やはり

- 新しい情報処理システムへの対応
- 機種、OSの違いを越えた支援ツール
- より使い易い支援ツール

などに対する要望は強い。

なお、この質問に対する回答の趣旨は、第6章「システム監査支援ツールの課題」に殆んど盛り込んであるので参考にされたい。

(2) 米国における現状*

① 環境変化への対応

この10年の間に、米国においてシステム環境が、バッチ処理からオンライン・データ捕捉、そしてオンライン・リアルタイム更新へとウエイトが移り、さらにパーソナル・コンピュータ、オフィス・オートメーションの急速な拡充とインフォメーション・センター構想の出現などをみるにいたっている。

現在、情報システム自体が4GL (Fourth Generation Language)、エキスパート・システムの採用、よりユーザ・フレンドリーなアプリケーションへと大きく変化するときにある。システム監査支援技法も当然このような影響を強く受けている。

② システムの高度化、複雑化への対応

「コンピュータを用いる情報システムが論理的にも技術的にも、増々複雑になるにしたがって、新しいEDP監査のツールが必要となる。利用できるツールや技法の数や種類にかかわらず、SRI (Stanford Research Institute) は、少ししかないと結論する。」これは1977年のSACレポートの結論である。

今回の調査結果より言えることは、基本的には、この状況は技術の発展が先行し、監査が後追いになることから変わらないにしても、当時、ITF法を理想的な方法としてSACレポートが述べたにもかかわらず、今回の調査対象会社は、ITF法は実務的な方法でないと結論を下している。

*：本調査は、今回、EDP監査人財団（米国）の協力を得て行ったものである。

これに対して、システム監査人にとってユーザ・フレンドリーな監査ソフトウェアの導入と開発を優先課題としているのが今回の調査の結論である。これがシステム監査人のニーズに適した方向である。

具体的には、メイン・フレーム上では、4GLなどの言語を利用したユーザ・フレンドリーな環境設定、パーソナル・コンピュータへのデータのダウンロードとユーザ・フレンドリーなパーソナル・コンピュータのソフトウェアの提供などがあげられる。これは、通常の業務処理のシステムと同一の傾向にあるといえる。

③ システム監査の生産性の向上

急速に変化する情報システムを適時に評価し、適格な意見を出すために、システム監査支援技法をより自動化していくことは重要な課題と考えている。このために、通常の監査業務をできるだけ自動化し、監査業務間のインタフェースを確立し、出力情報も異常なトランザクション、全体的な趨勢を示すものに限定しようとする傾向にある。このことにより、監査人を判断業務に特化させ、監査の生産性を向上させることができる。このためにエキスパート・システム、DSS(Decision Support System)の利用などが考慮されている。

また、監査ソフトウェアとは別にテスト・データ法については、開発担当者の作成したテスト・データを独立的にレビューすることによって監査人の作成したテスト・データと同様に考える方法が示されている。これは、セキュリティのコントロールなどにも利用しうる考え方であり、本来の監査機能の独立性からみても、自ら何らかのダミーデータなどを作成することには親しまないと思われるので、生産性の向上だけではなく、独立性の維持の観点からみても望ましい方向であるといえる。

④ 新たな監査技法の出現

1977年のSACレポートでは以下のような28の監査技法を紹介している。

監査の計画と管理

監査領域選択法

シミュレーション／モデリング法

得点法

複数現場監査用ソフトウェア法

管轄センター法

コンピュータ適用業務プログラムの統制のテスト

- テストデータ法
- 基本事例システム評価法
- 平行オペレーション法
- 総合テスト法（ITF）
- 平行シミュレーション法

データ処理トランザクションの選択と監視

- トランザクション選択法
- 組込み監査データ収集法
- 拡張レコード法

データの検証

- 汎用監査ソフトウェア法
- 端末監査ソフトウェア法
- 特殊目的監査プログラム法

コンピュータ・プログラムの分析

- スナップショット法
- 手作業によるトレース法およびマッピング法
- コンピュータ援用のトレース法およびマッピング法
- 統制流れ図法

コンピュータ・サービス・センター

- ジョブ会計データ分析法
- 監査手引き法
- 災害テスト法

適用業務システム開発

- 導入後監査法
- システム開発時点統制ガイドライン法
- システム開発ライフ・サイクル法（SDLC）
- システム検収および統制グループ法
- コード比較法

この中でシステム監査支援技法に相当するものは、○印が付されたものである。今回の調査でも多くの技法はここに挙げられたものと同一で

ある。これは、自動化の程度は別にしてシステム監査の機能自体に大きな変化がない限り、機能的に分類されたこれらの技法にも大きな変化があるとは思われないことから当然であるといえる。

10年前のシステム環境と現在のシステム環境を比較したときに大きな変化がもたらされている分野が2つある。一つはパーソナル・コンピュータを中心としたOA機能の発達、もう一つは、コミュニケーション・ネットワークの発達である。

これらの分野の発達がシステム監査支援技法に以下のような変化を与えている。

イ. 論理的アクセス管理（セキュリティ）を重視したソフトウェアの出現

これはSACレポートでも、一部すでにジョブ会計データ分析法として取りあげられているものの、オンライン・システムを直接的に対象としたものではなかった。今回の調査では、オンラインのシステム・ログやパラメータなどを直接的にレビューするソフトウェアが明らかにされた。

ロ. セキュリティ・ソフトウェアのユーティリティの利用

これもオンライン・システムの発達にともなってオープン・システムの論理的アクセスを管理するためのセキュリティ・ソフトウェア（RACF, ACF2, Top Secret など）が多くの企業に導入されていたことにより、これらの機能を監査にも活用しようとするものである。なお、10年前にはこれらのセキュリティ・ソフトウェアはほとんど普及していなかった。

これらのソフトウェアはバージョン・アップによって改善されているものの、まだ、一般の監査人に容易に理解できるものになっていない。

ハ. パーソナル・コンピュータを利用した監査ソフトウェアの出現

これもSACレポートに示されている特殊目的監査プログラム法と機能的には大きく変わるものではない。しかしながら、10年前には経済性の観点より、この方法はあまり普及しないと予想していた。現在では、すでに述べたようにメイン・フレームより監査目的のために選択されたデータをダウンロードさせ、これに対してそれぞれの特珠目的にあったソフトウェアをきめ細く対応させるために開発している。

このなかにはエキスパート・システムを用いたよりユーザ・フレンドリーなものも出現している。

また、パーソナル・コンピュータの機能を用いて内部統制質問書やリスク分析を効率的に行うことも考慮されている。

⑤ 実証性テストを重視した監査アプローチの増加

企業規模、取引量の拡大、コンピュータ・システムの出現などにより、内部統制の準拠性に重点を置いた監査アプローチが数多く採用されてきた。しかしながら、ここにきてシステム・ソフトウェアの活動をもその記録されているログを監査ソフトウェアを用いて実証的に監査しうることから、実証性監査に重点を移してくる傾向がみられる。調査対象となっている1社では、コントロールの準拠性テストは実施していないと言い切っている。これに代えて、システム・ソフトウェアの活動を分析する監査ソフトウェアを大幅に導入している。

⑥ 今後の我国のシステム監査への指針

我国のシステム監査を効果的、効率的に実施するためには、本報告書のC社の例が多く示唆を与えていると思われる。すなわち、多くトランザクション、システム・ソフトウェアなどの活動を実証的に分析し、異常な活動に焦点を当てた異常点監査手法の開発が望ましいものと思われる。

しかしながら、これらのソフトウェアの開発の前提として当然ながら内部統制、業務の側面よりみて何が異常であるか目的仕様を明確にすることが重要である。こういった意味から内部統制、および業務に対するより深い理解が必要となる。これは、開発時における詳細テスト・データをテスト・データとして監査用にも用いるために、より深い業務の理解が必要であると紹介されているところと合せて興味のあるところである。

このことから、システム監査支援技法の活用のためには、単にユーザ・フレンドリーな監査用ソフトウェアの開発だけでなく、業務（アプリケーション、システム開発、運用を含めた）の理解、内部統制、効率性・信頼性・安全性のガイドとなるビジネス・モデルの開発が必要であるといえる。

1.2.3 情報システム開発技法とシステム監査

(1) 開発管理と監査

従来、情報処理システムの信頼性や安全性を監査するには、完成したシステムを対象に、テストデータ法や平行シミュレーション法等を用いて診断し、概して部分的なチェックを行う方法が主としてとられて来た。

一方、ソフトウェアの開発側に於ては、1970年代から、ソフトウェア工学の概念が普及し始め、要求仕様定義から基本設計、詳細設計、プログラム設計、プログラミング、テストなどの各開発段階毎に管理点を設定し、品質管理を始めとするレビューを行い、ある段階の不備や誤りが、次の段階に波及しない様、十分な検査を行うことが推奨されるようになった。

いずれにしても、ソフトウェアというものの品質は、作成過程で組み込まれることから、出来上がったものを評価するのでは遅過ぎる。開発過程での幾つかの段階での介入が必要不可欠である。また、不可視性の極度に強いソフトウェアの実体をいささかなりとも目に見える状態にするものが作成過程で作成される種々のドキュメントであり、各段階毎にそれらドキュメントを含めて十分な検査を行うこととなる。

システム監査技法におけるSDLC (System Development Life Cycle)法は、予め設定した介入点毎に監査を実施する方法であり、当然ながら開発側での段階毎のレビューに共通する考え方である。立場上、開発側と監査側とのチェックポイントは若干異なるとはいえ、介入のタイミングは一致させることは可能であろう。

(2) コントロールと監査

コントロールなきところに監査なし、という言葉がある。従来の情報処理システムやソフトウェアの開発業務は、コントロール不在と言うに近いケースが多々みうけられた。

開発管理基準も、開発作業標準も存在せず、それに代るコントロールも存在しない環境で作成されたソフトウェアは、元々、監査の対象とはなり得ないというのがシステム監査の基本的立場のようである。したがって、監査可能な状態に持って行くだけでも、現在のソフトウェア開発業務はかなりレベルアップする筈だという意見も出てくる。

いずれにしても、各開発段階での作業内容、用いられる開発技法やツール、ドキュメントの記述法や内容等を始め、いわゆる開発作業標準の設定

と、各段階のレビューを含む開発管理基準の設定が先ず前提であり、監査の第一歩は、それらに準拠して業務が遂行させているか否かの監査に始まる。

(3) 管理技法と監査技法

現在ソフトウェアの管理技法は、開発技法に比べてもさらに遅れている。したがって監査側が開発側の管理技法やツールを流用することもむずかしい状態にある。

そこで以下のような機能が実現すれば、開発側、監査側双方に貢献することとなろう。開発側は、それによって開発作業の生産性や開発ソフトウェアの品質が向上し、監査側は対象ソフトウェアの可視性が向上することにより監査作業の精度を高めることができる。

① 資源管理との連動機能

例えばDD/DSを用いた開発技法

② 高水準の要求記述や設計記述手法

4GLや簡易言語に準ずる上流工程の記述法

③ 監査機能を包含する言語体系

システム自体に監査機能の組込みを可能とする

④ ロジックの可視化機能

ソフトウェアの内部ロジックを分析し、わかり易く表現しうる機能

⑤ 統合的開発技法の一要素としての監査機能

要求仕様化からプログラミングへの一貫した開発技法に監査機能を内蔵あるいは直結することにより、Auditability(監査可能性)を大幅に向上させたソフトウェアの体系的な開発が可能となる。

1.2.4 システム監査支援ツールの課題

内外におけるシステム監査支援ツールの現状調査や実験評価の結果をもとに、現在の支援ツールの課題をまとめた。また、それらの課題をふまえて、今後新たに開発すべき支援ツールの例示を行った。

現在の支援ツールの課題として、主な指適は、以下のようなものである。

- ① 近年、情報処理技術はめざましく進展し、情報処理システムの高度化、多様化などの変化は急速である。しかしながらシステム監査技術の進歩は遅く、システムの変化に全く追従していない。
- ② システム監査の担当者は、情報処理技術の専門家ではない可能性が高く、

特にブラックボックス性の強い情報処理システムの監査のための強力で且つ使い易い支援ツールに対する期待が大きい。

- ③ システム監査に用いるハードウェア・リソースの自由度を上げるために、パーソナル・コンピュータを含む小型機による支援ツールの開発が望まれる。
- ④ 機種やOSの差異に対処しうる移植性の高い支援ツールや監査ソフトウェアが望まれる。
- ⑤ 汎用監査ソフトウェアに加えて、業種別の監査ソフトウェアの出現も期待される。
- ⑥ リスク分析のための技法の確立や、支援システムの出現が望まれる。
- ⑦ 監査人にとって使用し易い、システムの効率評価のための支援ツールが必要である。
- ⑧ 情報フロー分析や、データベース・セキュリティ評価のための安全性監査支援ツールを充実すべきである。
- ⑨ オーディット・オートメーションのための監査業務の効率化、半自動化の支援が必要である。
- ⑩ AI技術を含む、知的な監査支援機能の出現が望まれる。
- ⑪ 監査支援システムやツールの経済的負担が大きい。

以上の諸課題をふまえた、新しい監査支援ツールの例を表1-1に示す。

表 1 - 1 新しいシステム監査支援ツールの例（その 1）

目	的	機 能
情報処理システムの変化への対応	データベースの監査	◦ ランダム・アクセスの動的な監査 ◦ DBMS 自体に内蔵された監査機能 ◦ オンライン端末からの監査 ◦ 第 4 世代言語への監査可能性の包含
	ネットワーク関連の監査	◦ 通信制御処理機能の監査 ◦ 通信プロトコルの標準や基準に対する準拠性監査 ◦ 通信ネットワークの障害対策能力の監査 ◦ 通信ネットワークのセキュリティ機能の監査 ◦ ネットワーク接続におけるゲートウェイ機能の監査 ◦ セキュリティ・レベルの異なるネットワーク間接続システムのセキュリティ管理機能の監査 ◦ 接続ホストのセキュリティ・レベルの差異に対するネットワーク管理機能の監査 ◦ 電子取引, E F T S などの管理 ◦ 分散処理システムのセキュリティの監査 ◦ O A システムのセキュリティの監査 ◦ 遠隔保守時における安全性監査 ◦ 監査用端末によるリアルタイム・モニタリング機能 ◦ 監査用トラフィック・ジェネレータやシミュレーション機能
	F A 関連システムの監査	◦ コントロール機能の監査 ◦ F A - L A N に接続された端末からの監査機能 ◦ マイコン内蔵機能の監査 ◦ ロボット制御機能の監査 ◦ システム内蔵の監査機能

表 1 - 1 新しいシステム監査支援ツールの例（その 2）

目	的	機 能
情報処理システムの変化への対応（つづき）	人工知能応用システムの監査	◦ エキスパート・システムの正当性の監査 ◦ システム自体に内蔵された監査機能 ◦ 人工知能機能を用いた支援ツール ・ 出力情報から入力情報を推論する診断エキスパート・システム ・ システム監査支援知識ベース・システム ・ アプリケーション理解のための人工知能システム ・ 監査情報分析用エキスパート・システム
ソフトウェア作成段階における監査支援ツール		◦ 開発段階毎（要求定義，基本設計，詳細設計，プログラム設計，プログラミング，テスト）のレビュー支援 ◦ モニタリング機能組込み支援 ◦ カバレッジ・テスト等におけるテストデータ・ジェネレータ ◦ ソフトウェア保守情報の統合的管理機能（ドキュメント自動作成機能を含む） ◦ プログラムにおける入力データ（トランザクション）検査機能の監査 ◦ ソフトウェア作成業務全体を通した一貫性のある統合ツール
システム監査業務の作業環境の改善	資源管理機能の利用	◦ 監査機能のマクロ・カタログ化 ◦ DD/DS を活用した支援ツール
	システム・ソフトウェア関連（基本ソフトウェア）	◦ システム・ソフトウェア自体の監査機能 ◦ システム・ソフトウェアに内蔵された監査機能 ◦ モニタ能力拡大のためのロギング・システム ◦ モニタ情報の保護能力向上と時系列把握機能 ◦ アプリケーションのモニタリング機能，半自動組込み支援 ◦ プログラム内の情報フロー分析機能

2. システム監査支援ツール・技法

2. システム監査支援ツール・技法

2.1 SACレポートによるシステム監査支援ツール・技法の概要

現在種々のシステム監査関連文献等で紹介されているシステム監査支援ツールや技法の殆んどは、1977年にIBM, SRI, IIAが協力してまとめたSACレポート^{*}が原典となっている。

SACレポートでは、当時この分野で先進的とみられた米国、カナダ、欧州、日本の約40の組織での考え方や経験によるシステム監査用のツールと技法に対する回答結果を、次の七つの視点で分類している。

- ① システム監査の計画を立てたり、作業の管理を行う時に用いた方法、考え方
- ② アプリケーション・プログラムの正確性をテストする時に用いた方法、考え方
- ③ 監査対象のトランザクションの選択と監視を行う時に用いた方法、考え方
- ④ データの正確性を検証する時に用いた方法、考え方
- ⑤ プログラムの分析をする時に用いた方法、考え方
- ⑥ コンピュータ・センターの運用の統制を検証する時に用いた方法、考え方
- ⑦ アプリケーション・システムの開発を監査する時に用いた方法、考え方

ここでは、SACレポートの分類に従って、28のシステム監査支援ツール・技法を簡単に整理して紹介する。なお、ツール・技法の名称は、必ずしもその内容を最適に表現しているとは言い難いが、SACレポートの日本語版に従った。

2.2 システム監査の計画と管理を行うためのツール・技法

ここに分類されているツール・技法は、次の五つがある。

- ① 監査領域選択法 (Audit Area Selection)
- ② シミュレーション／モデリング法 (Simulation/Modeling)
- ③ 得点法 (Scoring)
- ④ 複数現場監査用ソフトウェア法 (Multisite Audit Software)
- ⑤ 管轄センター法 (Competency Center)

^{*} System Auditability & Control, Data Processing Audit Practices Report

(1) 監査領域選択法

監査領域選択法は、複数の事業所を持っている企業で、監査する対象をどの事業所にするかを決定する時に適用する手法である。

〔目的〕・問題が起り得る領域に正確に狙いをつける

・情報システム監査の資源を最適（有効）利用する

〔機能〕・事業所の状況と業績を評価するためのキーとなる財務及び運用上の指標情報をマトリックス形式で提供する

☆指標の例：・期限の過ぎた売掛金

・延滞した出荷

・特定トランザクションの誤り率

・現金取り扱い業務の処理時期

〔要件〕・キーとなる指標データを提供するために、各種アプリケーション・システムからデータを集めるシステムを持っていること

〔形態〕・コンピュータ利用

(2) シミュレーション／モデリング法

シミュレーション／モデリング法は、従来からの監査技法である勘定分析の手法等を応用して作成したモデルを用いてシミュレーションし、論理的に関連した勘定バランスの予測値を求めて実績値と比較して、その間の相違を発見するために適用する手法である。

〔目的〕・より詳細な調査を必要とする分野の発見

〔利点〕・予測（見積）に用いる変数を多く取れる

〔欠点〕・シミュレーションに用いられる変数の特徴や変数間の関係が変化するとモデルの手直しが必要になる

・モデルの作成に時間と費用がかかる

・専門知識が要求される

〔形態〕・コンピュータ利用

(3) 得点法

得点法は、アプリケーション・システムのリスクを客観的に把握して監査の対象システムを選択するために用いる手法である。

〔目的〕・監査対象アプリケーション・システムの選定

〔機能〕・リスクを把握する特性毎に得点を求め、特性間の重みによってシステムの総合得点を提供する

☆特性の例：・システムの対象業務

- ・コントロールされている資産の量
- ・システム形態（バッチ、オンライン）
- ・データ・ファイルの数
- ・出力レポート数
- ・プログラムの数 等々

〔要件〕・得点法のアルゴリズムは、特性間の重み等を含めて検証し、繰り返し改善しなければならない

〔形態〕・コンピュータ利用／技法

(4) 複数現場監査用ソフトウェア法

複数現場監査用ソフトウェア法は、企業内の複数の事業所に設置されたコンピュータ・センターのアプリケーション・システムの監査を、中央で購入又は作成した監査用ソフトウェアを用いて各事業所で行うための方法である。

〔目的〕・標準化とコントロールの強化

〔機能〕・監査用ソフトウェアの基本的機能は、他の監査用ソフトウェアと変らない

〔利点〕・一つの監査ソフトウェアを用いることによって、標準化とコントロールの強化が図れる

〔欠点〕・各事業所の特殊処理は、対象外となる

- ・監査ソフトウェアの利用のために、多くの内部監査人の教育・訓練が必要になる
- ・監査ソフトウェアを購入又は作成するために、各事業所の環境に精通した中央側のスタッフが必要になる

〔適用〕・各事業所間のシステムが類似していること（類似しているほど効果が大きい）

☆類似の項目：

- ・ハードウェア
- ・OS
- ・アプリケーション・システム（機能、プログラム構造）
- ・ファイル（項目、レコード形式）

〔形態〕・技法（考え方）

(5) 管轄センター法

管轄センター法は、複数現場監査用ソフトウェア法とは逆に、監査用ソフトウェアを管轄（中央）センターで実行する方法である。

〔目的〕・監査の標準化とコントロールの強化

〔利点〕・分散したコンピュータ・センターの能力不足を補完できる

・監査の労力を節約できる

・一つの監査ソフトウェアを用いることによって、標準化とコントロールの強化が図れる

・監査ソフトウェアの利用のために、多くの内部監査人の教育・訓練が省ける

・監査が管轄センターで行われるので、スケジュール管理が容易になる

〔欠点〕・管轄センターと分散したセンターとの間でのデータの授受のために余計な時間がかかり、持ち時間が多くなる

・分散センターに大規模なデータベースを使用している時は、そのまま複写することが困難なために、分散側で必要なデータを抽出しなければならない

〔形態〕・技法（考え方）

2.3 アプリケーション・プログラムの正確性をテストするためのツール・技法

ここに分類されているツール・技法は、次の五つがある。

① テストデータ法（Test Data Method）

② 基本事例システム評価法（Base Case System Evaluation）

③ 平行オペレーション法（Parallel Operation）

④ 総合テスト法（Integrated Test Facility）

⑤ 平行シミュレーション法（Parallel Simulation）

(1) テストデータ法

テストデータ法は、監査のために用意したデータ（テストデータ）によって監査対象のプログラムを実行し、その結果と予め想定しておいたプログラムの処理結果とを比較して、プログラムの処理の正確性を検証するものである。

〔目的〕・本番プログラムの処理ロジック、計算及び統制を検証

〔利点〕・客観的な証拠が入手できる

〔欠点〕・有効なテストデータを作成するには、対象システムについての詳細な知識が必要

・テストデータの作成に膨大な時間が必要

〔適用対象〕・バッチ及びオンライン環境のプログラム

〔動作種別〕・本番プログラムを用いる

〔適用時期〕・本番プログラムの運用時以外

〔形態〕・技法

(2) 基本事例システム評価法

基本事例システム評価法は、アプリケーション・システムの処理の正確性を検証するための技法であり、テストデータ法的一种である。この技法は、テストデータとして、基本事例と呼ばれる標準的なデータの集合、即ち当該アプリケーション・システムの機能を網羅的に確認するための判定基準のデータ群を用いる。

〔目的〕・本番プログラムの処理ロジック、計算及び統制の検証

〔利点〕・客観的な証拠が入手できる

・ユーザ自身が基本事例を作成し、システムの開発・保守の作業に参加するために、開発・保守の誤りが早期に発見できる

〔欠点〕・システムの開発・保守に余分な費用がかかる

・当該システムに關与する広範なユーザの参画が必要となる

〔適用対象〕・バッチ及びオンライン環境のプログラム

〔動作種別〕・本番プログラムを用いる

〔適用時期〕・本番プログラムの運用時以外

〔形態〕・技法

〔補助手段〕・テストデータを効率的に作成するために、次のツールを用いると良い。

＊テストデータ・ジェネレータ

＊テスト・カバレッジ測定ツール

(3) 平行オペレーション法

平行オペレーション法は、保守（更新）したプログラムの処理の正確性を検証するための技法であり、保守（更新）前後のプログラムを同一データ（通常、本番データ）で処理して、処理結果を比較することによって行う。データにテストデータを利用すれば、テストデータ法的一种とも言える。

〔目的〕・保守（更新）後の本番プログラムの処理ロジック、計算及び統制の検証

〔利点〕・客観的な証拠が入手できる

・処理結果を比較することによって、比較的簡単に両者の違いが見つけられる

〔適用対象〕・バッチ及びオンライン環境のプログラム

〔動作種別〕・本番プログラムを用いる

〔適用時期〕・本番プログラムの運用時

〔形態〕・技法

(4) 総合テスト法

総合テスト法（ITF）は、アプリケーション・システムの正規の運用環境の下で、本番データの中にテストデータを混在させて、システム機能の正確性及び完全性及び安全性を検証する技法である。これも、テストデータ法の一つである。

〔目的〕・本番システムの正確性、完全性、安全性の検証と評価

〔利点〕・プログラムの機能のみならず、トランザクションの発生から入力の作成、出力の取り扱い等の人手の部分も含めて包括的なテストができる

・正規の運用に影響を与えないでテストできる

・正規の運用に組み込んで実施されるので、通常の運用費用に多少の費用が加わる程度である

〔欠点〕・テストデータを処理結果から取り除くための機構を開発し組み込むために、相当な時間と費用がかかる

・当該システムの処理結果から、ITFのテストデータを取り除くための特別な機構が必要

・上記のような機構を組み込むことによって、本番システムに影響を及ぼす可能性があり、影響が無いことの検証が必要である

〔適用対象〕・オンライン環境のシステム（特に、システムの運用を中断できないような24時間運転のシステム）

例：・請求業務システム

・売掛金／買掛金システム

・給与システム

〔動作種別〕・本番プログラムを用いる（ITFのための機能を組み込む）

〔適用時期〕・本番プログラムの運用時

〔形態〕・技法

〔補助手段〕・エラー等が発見されたときに、その原因を探るためのデータを収集する機能を組込んでおく（例えば、組込み監査データ収集法（2.4の②参照）の併用）

(5) 平行シミュレーション法

平行シミュレーション法は、本番プログラムと同様の機能、又はその一部の機能を実現した別のプログラム（シミュレーション・プログラム）と本番プログラムによって、本番データを平行処理し、その結果によって本番プログラムの正しさを検証するための技法である。

〔目的〕・計算及び処理ロジックの正確性の検証

〔利点〕・本番で処理されるのと同じ全てのトランザクション（データ）がテストされる

・テストデータの作成作業が不要になる

〔欠点〕・シミュレーション・プログラムの開発（保守）のために時間と費用が必要になる

・シミュレーション・プログラムの正しさを検証しておかなければならない

〔適用対象〕・バッチ／オンライン環境のプログラムの一部の機能

〔動作種別〕・本番プログラム／シミュレーション・プログラム

〔適用時期〕・本番プログラムの運用時

〔形態〕・技法

2.4 監査対象のトランザクションの選択と監視を行うためのツール・技法

ここに分類されているツール・技法は、次の三つがある。

① トランザクション選択法（Transaction Selection）

② 組込み監査データ収集法（Embedded Audit Data Collection）

③ 拡張レコード法（Extended Records）

(1) トランザクション選択法

トランザクション選択法は、本番アプリケーション・システムへの入力（トランザクション）から、内部監査人の指定した基準に合致したものを抽出す

るものである。

〔目的〕・内部監査部門が、問題の発生し得る領域に目を向けるための基礎データとなるトランザクションを選択する。

・ユーザ部門で起こり得る問題の識別や内部監査業務の計画の援助

〔利点〕・選択にコンピュータ・プログラムを利用すれば、選択時間が軽減する

〔欠点〕・選択のためのコンピュータ・プログラム開発の費用と時間が必要

〔適用時期〕・本番プログラムの運用時

〔形態〕・コンピュータ利用

(2) 組込み監査データ収集法

組込み監査データ収集法は、トランザクション（データ）収集プログラム・モジュールを本番用アプリケーション・プログラムに組込んで、運用時に入力トランザクションや本番プログラムが生成したデータを抽出する技法である。SCARF（System Control Audit Review File）やSARF（Sample Audit Review File）の名称で紹介されているものも、この技法と同じである。

〔目的〕・処理結果を監査するためのデータ（トランザクション）の収集
・処理手順の検証

〔機能〕・組込みモジュールの機能は次のとおりである

＊設定条件（収集基準）に従ったトランザクションの抽出

＊トランザクションの傾向分析

＊例外トランザクションの報告

〔利点〕・全トランザクションに対して連続的にモニタ（監視）できる

・大量のトランザクションに対応できる

・例外条件に対する全てのトランザクションを監視できる

〔欠点〕・組込みモジュールの開発、保守に費用がかかる

・収集されるデータの量が多すぎて、収集後の内部監査人の処理が大変である

・対象アプリケーション・システムの開発と連動してトランザクション（データ）収集プログラムを開発し、組込まなければならない

〔適用対象〕・オンライン環境のプログラム

〔適用時期〕・本番プログラムの運用時

〔形態〕・コンピュータ利用

〔補助手段〕・収集したデータが、ファイルに出力される場合は、その内容を編集（並び換え、フォーマット変換等）したりするために、汎用監査ソフトウェア（2.5の(1)参照）を併用するとよい

(3) 拡張レコード法

通常、アプリケーション・システムは、トランザクションの処理を行うと、その結果はいくつかのファイルに処理の段階毎に分かれて記録される。そのため、トランザクション処理の監査をする場合には、その全てのファイルを順に追跡して必要なデータを収集しなければならない。拡張レコード法は、本番用アプリケーション・システムに組込んだプログラム・モジュールによって、個々のトランザクション処理に関連（影響）するデータを集め、通常取引レコードに監査証跡として追加（監査証跡が追加されたレコードを拡張レコードと言う）する技法である。

〔目的〕・処理結果を監査するためのデータ（監査証跡）の収集

・トランザクションに対する処理手順や規準の準拠性のテスト

〔機能〕・組込みプログラム・モジュールの機能は次のとおりである

＊データの収集

＊収集データ（拡張レコード）のファイリング

〔利点〕・本番システムを運用することによって、監査証跡が作成される

〔欠点〕・組込みプログラム・モジュールのために、本番システムの運用にオーバーヘッドがかかる

・本番用システムのみが必要なレコードよりも、レコードがかなり拡張され、そのためにより多くの記録媒体が必要になり、また費用も発生する

・組込みプログラム・モジュールの開発、保守に費用がかかる

〔適用対象〕・バッチ環境のプログラム

例：リスクの高い会計システム

〔適用時期〕・本番プログラムの運用時

〔形態〕・コンピュータ利用

2.5 データの正確性を検証するためのツール・技法

ここに分類されているツール・技法は、次の三つがある。

- ① 汎用監査ソフトウェア法 (Generalized Audit Software)
- ② 端末監査ソフトウェア法 (Terminal Audit Software)
- ③ 特殊目的監査プログラム法 (Special-Purpose Audit Programs)

(1) 汎用監査ソフトウェア法

汎用監査ソフトウェア法は、コンピュータ・プログラム群から成る汎用の監査ソフトウェア・パッケージを使用して、アプリケーション・システムで持っているファイル进行操作して、監査に必要なレポートを作成する技法である。

〔目的〕・監査に必要なデータ（証拠）を収集する

〔機能〕・全ファイルまたは一つのファイルからデータを抽出して、その合計、クロスの合計

- ・データに対して各種の論理演算を施しレポートを作成する
- ・ファイル中のデータの選定とレポート作成
- ・ファイル中のデータの層別化
- ・ファイル中のデータの統計的サンプリング
- ・ファイルのコンペア

〔利点〕・使用するためにコンピュータの特別な知識は要らない

- ・長年使い込まれ、その正確性が立証されている
- ・汎用的なパッケージであるために、多くのアプリケーション・システムに適用できる

〔欠点〕・現状では利用できるコンピュータが限定されている場合が多く、特に小型コンピュータでは適用できない

- ・対象ファイルがデータベース化されていると適用出来ない場合が多い

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・本番プログラムの運用と切り離して適用

〔形態〕・コンピュータ利用（パッケージ化されたツール）

(2) 端末監査ソフトウェア法

端末監査ソフトウェア法は、端末装置からオンライン問合せ用の言語（命令）を用いて、データベースにアクセスし監査に必要なデータを収集する技

法である。

〔目的〕・監査に必要なデータ（証拠）を収集する

〔機能〕・汎用監査プログラムと同様の機能を持っているが、使用するオンライン問合せ用の言語によって異なる

〔利点〕・素早くデータが収集出来る

・収集結果を見て、その場で他のデータが必要であるか等の検討が出来る

・データベースが運用状態になっていれば、何時でも利用できる

〔適用対象〕・オンライン環境のシステム

〔適用時期〕・本番プログラムの運用と切り離して適用

〔形態〕・コンピュータ利用（ツール）

(3) 特殊目的監査プログラム法

特殊目的監査プログラム法は、特定のアプリケーション・システムのファイルから監査データを収集するために、特別に作成したプログラムを用いて行う技法である。

〔目的〕・監査に必要なデータ（証拠）を収集する

・汎用監査ソフトウェアで不十分なところをカバーするとき、汎用監査ソフトウェアが無いときに適用する

〔利点〕・監査目的に合わせたデータが、監査人の要求する形式で提供できる

〔欠点〕・改めてプログラムを作成しなければならず、作成費用がかかる

・特定の目的であるから、柔軟性がない

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・本番プログラムの運用と切り離して適用

〔形態〕・コンピュータ利用（ツール）

2.6 プログラムの分析をするためのツール・技法

ここに分類されているツール・技法は、次の四つがある。

① スナップショット法（Snapshot）

② トレース法（Tracing）

③ マッピング法（Mapping）

④ 統制流れ図法（Control Flowcharting）

(1) スナップショット法

スナップショット法は、アプリケーション・プログラムの中で処理しているデータの内容（メモリに入っている）を、指定したタイミングで取り出し、表示するための技法である。通常は、プログラム作成時のデバックに使用される。

〔目的〕・データ（トランザクション）の処理の過程をレビュー

〔利点〕・アプリケーション・プログラム中の処理の異常を分析するデータを取得できる

〔欠点〕・プログラムの処理過程を理解していないと、スナップショット点が決められない

- ・使用するのにコンピュータの知識が必要

- ・必要の都度、アプリケーション・プログラムに組込まなければならない

〔適用方法〕・スナップショット・ルーチンを本番アプリケーション・プログラムに組込んで使用する

- ・組込み場所は、監査人が表示したいデータの処理過程であり、通常、次のようなタイミングの場所がスナップショット点となる

- * トランザクションがプログラムに入力された時

- * トランザクションが複雑な処理ルーチンを終了した時

- * 処理の判断が行われる時

- * 処理の結果がファイルに出力される直前

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・本番プログラムの運用時またはそれ以外

〔形態〕・コンピュータ利用（ツール）

(2) トレース法

トレース法は、処理中のトランザクションの流れを追跡（トレース）する技法である。これも通常は、プログラム作成時のデバックに使用される技法である。

〔目的〕・プログラムの処理が定められた手順通りかを判断する

〔利点〕・トランザクション対応に処理の順序が明確になる

〔欠点〕・プログラムの実行に、かなりのオーバーヘッドがかかる

- ・プログラムの処理内容が理解できていないと利用できない
- ・本番運用時には、トレースの指示をプログラムから取り除かなければならない

〔適用方法〕・監査対象のプログラムが記述されているプログラム言語のオプション機能で指示することによって、トレースができる

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・本番プログラムの運用時以外

〔形態〕・コンピュータ利用（プログラム言語のオプション）

(3) マッピング法

マッピング法は、アプリケーション・プログラムのステートメントが実行されたかどうか、また何回実行されたかを確認するための技法である。これもプログラム作成時のデバックに利用される技法である。

〔目的〕・プログラム中の実行されない部分を探し出す

〔利点〕・プログラム中の実行されない部分が明確になることによって、その部分を取り除き、コンピュータ資源（ここでは、記憶装置）の使用効率をあげることができる

〔欠点〕・プログラムが正しい順序で実行しているかは、判断できない

〔適用方法〕・プログラム分析パッケージを使用する。このパッケージが測定する主な結果は次のとおりである

- * 実行されないプログラム・ステートメント
- * 各プログラム・ステートメントが実行された回数
- * 各プログラム・ステートメントの実行に要した時間

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・本番プログラムの運用時以外

〔形態〕・コンピュータ利用（ツール）

(4) 統制流れ図法

統制流れ図法は、流れ図（フローチャート）によって、内部統制の状況を表現する技法である。流れ図には、レベルによって、企業全体を捉えて主要機能（業務）とその間の統制を表すもの、主要機能（業務）に注目して処理とその間の統制の状況を表現するもの及び処理に注目してそれを更に分解した要素で統制の状況を表わすもの等がある。

- 〔目的〕 ・ 統制の適切さの評価
 - ・ システム変更が統制に及ぼす影響のチェック
- 〔利点〕 ・ 統制状況が理解できる
 - ・ 重要な統制の欠落が発見できる
 - ・ 文章と違って、理解しやすい
- 〔欠点〕 ・ 流れ図の修正に手間がかかる
- 〔適用段階〕 ・ 予備調査
- 〔形態〕 ・ 技法

2.7 コンピュータ・センターの運用の統制を検証するためのツール・技法

ここに分類されているツール・技法は、次の三つがある。

- ① ジョブ会計データ分析法 (Job Accounting Data Analysis)
- ② 監査手引法 (Audit Guide)
- ③ 災害テスト法 (Disaster Testing)

(1) ジョブ会計データ分析法

ジョブ会計データ分析法は、OS (オペレーティング・システム) の機能の一部であるジョブ会計データ収集機能が収集したデータを監査に利用しようとする技法である。

- 〔目的〕 ・ コンピュータ・センター運用の監査
- 〔利点〕 ・ データ資源利用についての情報が経済的に入手できる
 - ・ コンピュータ資源の管理、データ・セキュリティの監査に情報が利用できる
- 〔欠点〕 ・ 収集した情報が大量になる
 - ・ 監査対象が収集した (操作して) 情報を監査に利用する (監査の独立性の問題)
- 〔適用方法〕 ・ OS (オペレーティング・システム) の機能の一部であるジョブ会計データ収集機能を利用して次の情報を収集する
 - * ジョブ開始、終了時間
 - * データ・ファイルの利用度
 - * アペンド・ジョブの一覧
 - * CPUの使用時間
 - ・ 収集された情報の中から、監査に必要なデータをプログラ

ムで抽出する

〔形態〕・技法

(2) 監査手引法

監査手引法は、内部監査人が、監査の実施手順や質問（チェックリスト）等を明らかにした手引（ガイドライン）を用いて監査する技法である。

〔目的〕・監査の実施を効率化する

・監査の手順等を標準化する

〔利点〕・不慣れな監査人でも一定水準の監査ができる

・監査の準備時間を減少できる

・同様な監査対象を均質的に監査できる

〔欠点〕・監査ガイドに依存しすぎると、重要なポイントが見逃される

・監査ガイドが、監査の対象に当てはまらないことがある

〔形態〕・技法

(3) 災害テスト法

災害テスト法は、災害時に行うべきと定めた対策が、どの程度有効に機能するかをテストすることによって、災害対策を監査する方法である。

〔目的〕・災害対策の有効性の確認

〔利点〕・災害対策の欠陥を発見できる

・災害対策の有効性を保証する

〔欠点〕・コンピュータ・センターの業務運用が中断する

・災害対策が有効に働かなかった場合、業務運用がストップする恐れがある

〔補助手段〕・災害対策の評価のために、得点法（2.2の(3)参照）を用いるとよい

〔適用対象〕・バッチ／オンライン環境のシステム

〔形態〕・技法

2.8 アプリケーション・システムの開発を監査するためのツール・技法

ここに分類されているツール・技法は、次の五つがある。

① 導入後監査法（Postinstallation Audit）

② システム開発時点統制ガイドライン法（Control Guidelines for Use During System Development）

③ システム開発ライフサイクル法 (System Development Life Cycle)

④ システム検収及び統制グループ法 (System Acceptance and Control Group)

⑤ コード比較法 (Code Comparison)

(1) 導入後監査法

導入後監査法は、アプリケーション・システムが開発を終了し、本番稼働環境に移された場合には、直ちに内部監査人がそのシステムを監査しなければならないとの考え方を示したものであり、監査の方法または技法を指すものではない。

〔目的〕・導入後監査の目的は、次のとおりである

- * システムが要求仕様を満足しているかの確認
- * システムが適切にコントロールされているかの確認
- * システム・ドキュメントが規準に合致しているかの確認
- * システムの扱うデータが完全でしかも妥当であるかの確認

〔利点〕・導入後監査を行う利点は、運用に先駆けて次の事が確認できる点にある

- * データが正確且つ安全であるということ
- * システムが組織の方針や手順を遵守していること
- * システムが目的を達成していること
- * アプリケーション・システムのコントロールの適切さ

〔欠点〕・システムの誤りを訂正する費用や時間の軽減には、あまり貢献しない

〔補助手段〕・監査対象を決定するために、得点法 (2.2 の(3)参照) を用いるとよい

- ・監査の手順は、監査手引法 (2.7 の(2)参照) を利用できる
- ・監査情報の収集に、汎用監査ソフトウェア法 (2.5 の(1)参照) が利用できる

〔適用対象〕・バッチ／オンライン環境のシステム

〔形態〕・考え方

(2) システム開発時点統制ガイドライン法

システム開発時点統制ガイドライン法は、当該システムの開発段階で内部

監査人が開発内容を分析して、最適な内部統制が組込まれているかを監査し、その結果を開発にフィードバックさせようとする考え方で、監査のときに内部統制ガイドラインを利用するものである。

〔目的〕・最適な内部統制の構築

〔利点〕・早期（開発時点）に内部統制の不備を発見できる

・そのために運用段階で発見した時よりもはるかに少ない労力と費用で、対処できる（即ち、導入後のシステム変更を回避できる）

〔欠点〕・内部監査人が、開発作業上の役割にのめり込む可能性がある

〔適用対象〕・バッチ／オンライン環境のシステム

〔形態〕・考え方

(3) システム開発ライフサイクル法

システム開発ライフサイクル法（SDLC）は、システム開発の過程をいくつかの段階に分割して、その段階をチェック・ポイントとして内部監査人がそこで作り出された物（設計書等）を観察・点検し、システムの正確性や統制の正当性等を確認しながら開発していこうとする考え方である。システム開発時点統制ガイドライン法（2.8の(2)参照）とほぼ同様の考え方であり、その違いは統制のガイドラインが用意されているか否かである。

〔目的〕・システム開発の不備を早期に発見すること

〔利点〕・費用や時間のかかる導入後のシステム変更を最小限に押さえることができる

〔適用対象〕・バッチ／オンライン環境のシステム

〔形態〕・考え方

(4) システム検収及び統制グループ法

システム検収及び統制グループ法は、システム監査人（内部監査人）が不得手とするシステム開発過程の点検（モニタ）を、EDP処理部門の要員で構成したシステム検収及び統制グループに分担させ、内部監査人はシステムの運用監査に重点を置こうという考え方である。

〔目的〕・コンピュータの知識とかなりの労力を要するシステム開発過程の点検から内部監査人を解放し、内部監査人の得意とする分野に力を注がせる

〔利点〕・内部監査人が開発業務に直接関係しないので、独立性が確保できる

・運用段階での監査実施費用が減少する

〔欠点〕・システム検収及び統制グループに監査の知識を習得させなければならない

〔適用対象〕・バッチ／オンライン環境のシステム

〔形態〕・考え方

(5) コード比較法

コード比較法は、アプリケーション・システムのプログラム・コードが正しく維持されているかをチェックするために、異なる時点で作成した（例えば、本番用のプログラムとそれをバックアップしたもの）プログラム・コード同士を比較する方法である。比較には、コンピュータを利用する。

〔目的〕・プログラム変更／保守手順及びプログラム・ライブラリ作成手順が正しく守られているかの検証

〔利点〕・プログラムの不正変更等が容易に発見できる
・費用も少なくすむ

〔適用対象〕・バッチ／オンライン環境のシステム

〔適用時期〕・プログラム変更／保守時

〔形態〕・考え方

2.9 システム監査支援ツール・技法の比較

以上紹介したシステム監査支援ツール・技法を、次の視点で整理して比較し、表2-1に一覧表にして示す。（但し、監査の計画等に用いる一部の技法は割愛した。）

(1) 何を対象として監査するか

- ① データ・ファイル
- ② アプリケーション・プログラム（AP）
- ③ コンピュータ・センター
- ④ オペレーティング・システム（OS）

(2) システム・ライフサイクルのどの段階の監査に適用するか

- ① 企画段階
- ② 開発段階
- ③ 運用段階

(3) 何を目的として監査するのか

- ① 準拠性

- ② 信頼性
- ③ 安全性
- ④ 効率性

(4) どのような事が出来るのか

- ① データの選択・監視
- ② データの分析
- ③ データの正確性の検証
- ④ プログラム機能の正確性の検査
- ⑤ プログラム・ロジックの分析

(5) ツール・技法の区分

- ① ツール化またはプログラム化されている

*汎用である

*専用目的である

*監査対象アプリケーション・プログラムに組込んで利用

*他と分離（独立）して利用

- ② 技法・考え方である

表 2-1 現在の代表的なシステム監査支援ツール・技法の例（その1）

比較の視点 技法・ツール	(1)監査対象				(2)使用段階			(3)監査目的				(4)機 能				(5)区 分					
	データ・ファイル	A P	コンピュータ・センタール	O S	企画 段階	開発 段階	運用 段階	標準 性	信 頼 性	安 全 性	効 率 性	データの選択・監視	データの分析	データの正確性の検証	プログラムの正確性の検査	プログラム・ロジック分析	ツールの				技法・考え方
																	汎用	専用	組込型	分離型	
① テストデータ法		○				○	○	△	○		△				○	△	△				○
② 基本事例システム評価法		○				○	○	△	○		△				○	△	△				○
③ 平行オペレーション法		○				○	○	△	○		△				○	△		○		○	
④ 総合テスト法		○				△	○	△	○	○					○	△		○	○		
⑤ 平行シミュレーション法		○				△	○	△	○		△				○	△		○		○	
⑥ トランザクション選択法	○	△					○					○	○	△				○		○	

表 2-1 現在の代表的なシステム監査支援ツール・技法の例（その2）

比較の視点 技法・ツール	(1)監査対象				(2)使用段階			(3)監査目的				(4)機能				(5)区分				技法・考え方	
	データ・ファイル	A P	コンピュータ・センター S	O	企画 段階	開発 段階	運用 段階	標準 性	信頼 性	安全 性	効率 性	データの選択・監視	データの分析	データの正確性の検証	プログラムの正確性の検査	プログラム・ロジック分析	ツール				
																	汎用	専用	組込型		分離型
⑦ 組込み監査データ収集法	○	△					○					○						○	○		
⑧ 拡張レコード法	○	△					○					○						○	○		
⑨ 汎用監査ソフトウェア法	○	△				△	○	△	○				○	○	△		○			○	
⑩ 端末監査ソフトウェア法	○	△				△	○	△	○				○	○	△		○			○	
⑪ 特殊目的監査プログラム法	○	△				△	○	△	○				○	○	△			○		○	
⑫ スナップショット法	△	○				○	○		○						○	○	○		○		
⑬ トレース法	△	○				○	○		○						○	○	○		○		
⑭ マッピング法		○				○	○		○						○	○	○			○	
⑮ 統制流れ図法		○			△	△	○	△	○						○	○					○
⑯ ジョブ会計データ分析法		○		○			○	○		○	△		△	△			○			○	
⑰ 災害テスト法			△	○	△		○	○		○											○
⑱ 導入後監査法		○	△				○					○					○			○	
⑲ システム開発時統制ガイドライン法		○			○	○		○	○	△	△				△	△					○
⑳ システム開発ライフサイクル法		○				○	○	○	○	△	○				○	○					○
㉑ システム検収・統制グループ法		○	△			○	○	○	○	○	○				○	△					○
㉒ コード比較法		○	△			○	○	△	△	○					○		○			○	

注) ○：対応する，△：対応することもある

3. システム監査支援ツールの現状調査

3. システム監査支援ツールの現状調査

今後開発すべきシステム監査支援ツールの要件に反映させる目的で米国及び日本の現状について調査を行った。

3.1 米国における現状

3.1.1 調査の概要

(1) 調査目的

本調査の目的は、米国における最新のコンピュータ支援監査技法 (CAAT: Computer Assisted Audit Techniques) の実態を把握し、わが国における今後のこの分野の指針を得ようとするものである。

(2) 調査方法

米国ではシステム監査支援ツールを使って本格的なシステム監査を実施している企業がかなりあるので、内部監査の一環として高度なシステム監査を実施している3社を選択し、訪問調査を行うことにした。

(3) 調査対象

本調査の対象となった事業体および内部監査組織の概要を示すと以下のとおりである。

① A製材会社

イ. 業務の概要

A製材会社は、米国デラウェアに本拠を置く国際的なコングロマリットで、21,000人の従業員をかかえ、世界9ヶ国で62ヶ所の拠点をもち、70万エーカーをこえる森林と6鉱山を有する。主たる業務を挙げると次のようになる。

- ・ファイバークラス製品部門……絶縁ファイバーおよび撚り糸状ファイバークラス繊維など
- ・森材製品部門……木材、合板、紙、パッキング製品、パルプなど
- ・特種製品部門……屋根材、ろ過用材、光製品、産業用絶縁材など、さらに、鉱物としては珪素、パーライトを産出

ロ. 情報システムの概要

米国内に3つのデータ・センターを有する。最大のセンターは、コロラド州デンバーにある世界本部にあり、以下のような機器構成を持

ち、集中型のアプリケーション・システムを運用している。

- ・ IBM 3084 MVS/XA
- ・ 8 テープ・ドライブ, 32 ディスク・ドライブ, 3 レーザ・プリンタ
- ・ 3 分散拠点
 - 他の2つのデータ・センター (IBM 43XX, DOS) とのリンク (DATAPOINT および WANG)
 - 2 分散拠点は, 世界中の拠点をネットワークで結ぶ (IBM 8100, WANG, DEC など)

ハ. 内部監査部とシステム監査担当者

・ 人員構成

内部監査部は, 31名の内部監査担当の職員によって構成され, 平均就業年数は12年をこえ, 平均監査就業年数もほぼ5年に達する。8名は公認会計士 (内2名は公認内部監査人) で, 3名は公認情報システム監査人 (CISA: Certified Information Systems Auditor) である。

・ 目的と組織

A製材会社の内部統制の有用性と効率性の向上と, 会社資産の適切な保全を確保することを目的として, 社内の業務を調査し評価することが内部監査部の役割である。監査の結果は, 監査委員会, 取締役会および各管理者に報告される。

内部監査部では, 現在以下の諸点を重視している。

- 取引および財務上の契約履行の約束の承認の条件
- 財務および業務情報の信頼性およびインテグリティ
- 方針, 計画, 法律, 規則への遵守を確保するシステムの確立
- 社内の重要な文書類および記録を含めた諸資産の保全の手段

② B銀行

イ. 業務の概要

B銀行は, ニューヨークに本拠を置く国際的な銀行で, 5つの事業部の独立採算による分散的な経営管理が行われている。これらの5つの事業部の提供するサービスの内容を示すと次のようになる。

- ・ 資産を基にした財産運用……不動産, 機器を対象とした資金提供
- ・ 国際サービス業務……州際間, 国際間の取引に対するサービス

・法人顧客サービス業務……信託、証券、情報提供をもふくめた伝統的かつ革新的サービス

・個人顧客サービス業務……一般大衆および小規模企業を対象とした預金、貸付業務

いずれの事業部においても、その業務の成果はマネジメント・インフォメーション・システムによって自動的に作成される。しかしながら、このシステムは事業部の利益向上のために絶えず変更されている。

ロ. 情報システムの概要

米国内には、6つの主たるデータ・センターと25の小さなセンターがある。また、世界的にはロンドンに主たるデータ・センターがあり、ヨーロッパの他の都市、アジア、南アメリカの支店には小さなデータ・センターがある。米国内における主たる業務別のハードウェア、システム・ソフトウェア及びセキュリティ・ソフトウェアの内容を示すと、以下のようになる。

業 務	ハードウェア	システム・ソフトウェア	セキュリティ・ソフトウェア
国際サービス業務	IBM3084	MVS, DOS, VM	ACF2, GUARDIAN
	IBM System 36	System 36 Support System	
	DEC1184	RSTS/E	
証券サービス業務	IBM3090	MVS	ACF2
	IBM3084		
法人顧客サービス業務	IBM3090	MVS	ACF2, GUARDIAN
	IBM3084		
個人顧客サービス業務	IBM4381	MVS, VM	ACF2
	NCR	VRX	
	PERKIN-ELMER	OS/32	
	TANDEM	GUARDIAN	
CIT業務	IBM4381	MVS, CICS, IDMS	Top Secret

なお、これらのデータ・センターは、グローバル・ネットワークによって接続されている。この結果、適時に、信頼できるグローバルな情報を作成することができる。このようなグローバルな金融システムを持っている銀行は現在世界中で5行あるのみである。

最近の傾向としては、大型のメイン・フレームよりミニコンピュータ、マイクロコンピュータを基礎とするシステムへの移行がみられる。

ハ. 内部監査部とシステム監査担当者

・人員構成

内部監査部は、313名の職員によって構成されている。ニューヨークに本部を置き、ロンドン、シンガポール、ロサンゼルス、アトランタに地域本部を有する。当部はさらに7つの部門に区分されている。プロフェッショナル業務部門と5つの事業部門、システム監査部門（愛称CATS: Computer Audit Assistance）からなる。

・目的と組織

内部監査部の目的は、次のことが適切に行われているか継続的に第三者の視点より評価し、社外重役のみで構成される監査委員会に報告することにある。

- 資産が適切に使用、保全されているか。
- 財務上のコントロールが整備され、適切に運用されているか。
- 法律、規則、方針、手続に従って適切に業務が実行されているか。
- 重要なコントロール上の問題点が発見されれば、すみやかに適切な措置がとられているか。

③ C製薬会社

イ. 業務の概要

世界の230カ所に拠点を持つ、医薬品、医療器具の製造販売を行っている高度に分散化された国際企業である。本社による管理はゆるやかで、各個所の独自性がかなり認められている。

ロ. 情報システムの概要

コンピュータ機器は、IBM製が95%を占める。IBM S/34より3090のシエラ・シリーズまでの種々の機器が利用されている。ソフトウェアは、主として社内で開発されており、人的資源管理システム以外には、共有される本社システムはない。それぞれの拠点において特有のシステムと手続を持つことがゆるされている。最近の傾向としてはソフトウェア・パッケージの利用が増加している。

ハ. 内部監査部とシステム監査担当者

・人員構成

当社の内部監査部は、組織的には社外重役で構成されるシステム監査委員会の下部機関である。監査報告書は財務部長(Treasurer)に提出される。内部監査部の社員は約90名であり、そのうちシステム監査の担当者は10～15名である。

・目的と組織

内部監査部は、マネジメントの支援を主たる目的として25年前に設立された。このために、定常業務の定期的監査、導入時の監査、特種目的の監査を実施するだけでなく、手作業と情報システムの内部統制のハンドブックを作成している。当部門の目的と方針をより具体的に示すと、以下ようになる。

- 3年に一度は全拠点を監査する。
- 業務処理とコンピュータ部門のコントロールの監査を統合的に実施する。
- 通常の監査人とシステム監査人の相互教育を実施する。
- 導入時の監査が可能であるならば、できる限り実施する。
- 監査スタッフのレベルを維持するために絶えず社内教育を実施すると同時に技術支援機能をそなえる。

3.1.2 A 製材会社

(1) システム監査の方針

内部監査部では、情報システムのインテグリティを確かめるために多大の監査資源を投下している。一般の内部監査人とシステム監査人は、チームを組んで情報システム監査の効率性を高めようとしている。このために、特に重要な財務報告の分野の内部統制、残高記録の正確性、およびトランザクション処理の有効性を重点的に監査している。

また、高度なコンピュータ支援監査技法を継続的に開発、利用し、次のような監査を効果的に実施するのも重要な課題である。

- ・重要なシステム処理の再実行
- ・重要な本番プログラムの変更の明確化
- ・データ・ファイルの正確性のテスト
- ・特定の目的のために会計取引のリストの作成

- ・会計記録の分析
- ・監査用統計的サンプリングの使用

(2) コンピュータ支援監査技法の概略

① 背景

過去12年の間に内部監査部では数百の監査用のソフトウェアを特定の目的のために開発してきた。しかしながら、高度で複雑なアプリケーション・システムの利用が広がるにつれて、他のコンピュータ支援監査技法を考慮する必要が生じてきた。

EDP内部監査の生産性を向上させることが、経営トップの期待に応え、適時に正確に取引の処理と社内の方針と手続の準拠の度合の評価を実施することにつながる。このために、内部監査の責任者は、重要な取引を常に監査する方法と、業務の趨勢をハイライトするための財務情報 요약の方法を考えてきた。

② 組込み監査技法およびITF法

このために考案された方法として本番用プログラムのなかに監査手続を組み込む技法がある。この技法を利用することによって、監査のためにすぐに取引を選択し、その合理性や社内の方針への準拠性をテストすることが可能となる。また、この方法を利用することによって事後監査のために取引を任意に抽出することもできる。

組込み監査技法の展開されたものとして、本番用プログラムのなかに監査プログラムを組み込む代りに総合テスト法(ITF)を利用することも考えた。この技法はミニカンパニー法ともいわれるものであり、架空の口座を設定し取引を受け付け、その出力を分析することにより、適切なデータ処理が実施されているか判定するものである。

これらの組込み監査技法およびITF法の両者にみられる主たる弱点は、アプリケーション・システムのなかに監査用のプログラムを組み込むことが要求されることである。監査用プログラムの組込みがシステムをより複雑にし、プログラムの保守時に情報システム部門に負担をかけることになる。さらに、これらの技法の最も大きな欠点は、情報システム部門の担当者に監査プログラムの組込み時、保守時に依存することになり、監査の独立性を阻害することにある。

③ 高度コンピュータ支援監査技法

このような背景および経験より、A製材会社では以下のような、より経済的に内部監査の目標を達成することを目的として高度コンピュータ支援監査技法を開発した。

- ・監査データベース機構法 (Audit Database Facility)
- ・継続監査法 (Continuous Auditing Procedures)
- ・プログラム変更コントロール法 (Automated Program Change Control Techniques)

(3) 高度コンピュータ支援監査技法の具体例

① 監査データベース機構法

イ. 目的

この技法は、内部監査人の業務の生産性の向上を目的として、SAS社によって提供されているエンド・ユーザ統合ソフトウェアSAS (Statistical Analysis System) を主に利用して構築されている。生産性の向上は、OS JCL, SASやTSO (Time Sharing Option) のコマンドの詳細な知識がなくても運用できる環境を用意することと、監査知識のライブラリを提供することで達成できる。

ロ. 構成

当技法の基本となるものは、ユーザの作業環境を生成するためのTSOコマンド・ランゲージ・プロセッサである。この基本となる部分を基本システム (Base System) と呼んでいる。

基本システムは、必要なファイルをコピーし、ユーザにファイルと作業環境を提供し、基本的な入力と出力を実行する。これらのジョブはSASマクロを利用して、実行される。

基本システムは、開発された監査手続の正確な再実行ができるように、それ自身が仕様のドキュメンテーションになるように構築されている。

当技法は、ユーザにオンラインで利用上必要な情報を提供している。多層化された使用方法の説明のヘルプ・コマンド、データベースの内容の説明、ユーザ間の情報交換のための伝言板 (Bulletin-board) などの情報がこれにあたる。

保守の容易性を確保するために、サービス (ヘルプ、ディレクトリ、報告書のプリントなど) 提供のための支援ファイルの構造をできるだ

け単純化した。このため基本的なTSOのエディタを通して容易に保守することができる。

ハ. 機能

当技法のなかで最も利用されているのは、メニュー（sic）システムである。このシステムでは、一般の内部監査人でも対話型で容易にデータを抽出、加工し、報告書を作成できるよう設計されている。ただしSASの標準フォーマットのデータ・セットは、システム監査人がコピーして作成するのが一般的である。しかしながら、SAS、TSOなどの知識がなくても、本番用のデータに独自にアクセスし、監査用のデータベースを作成できる。これはメニューによって起動されるプロンプトを活用することによって可能となった。

標準フォーマットのデータ・セットは、翌週の監査の作業スケジュールに基づいて毎週末にシステム監査人によって作成される。

例えば、あるファイバークラスの工場に往査する計画があるとするならば、週末に次のようなデータベースが作成される。

- ・最新の月次試算表（詳細）
- ・最近2年間の当工場の元帳のデータ
- ・最近2年間の当工場の買掛金と支払のデータ
- ・最近2年間の売上および売上原価、製造原価のデータ

など

このデータベースに対して、メニュー・システムを利用して実行できる機能としては次のようなものがある。

- ・システムのプロンプトとして作成されたパラメータを利用してソートする。
- ・データをレビューする。
- ・印字出力するデータ項目を選択する。
- ・データ項目の合計を計算する。
- ・金額をもとにしたデータを選択する。
- ・データ項目の集計をする。
- ・データ項目に対して計算を行う。
- ・上述の機能を組合せ、実行する。
- ・報告書をプリントする。

これらの諸機能によって作成されたファイルは、その内容をオンラインで確かめた後、印字出力される。また、このファイルを用いてさらに他のデータベースやファイルとの追加的処理を行うこともできる。

本技法では、基本システム以外に、習熟した監査人（SASユーザ）のために、以下のような特殊技法を用意している。

- ・自動監査調書（Automated Work Papers）

SASのワード・プロセッサ機能であるFSLETTERを用いて、監査部用のワード・プロセッシング用データベースと各担当者別用のデータベースを用意する。自動的に作成された監査調書はオンラインでレビューした後、将来参照するために保管される。

採用された監査ソフトウェアのロジックの永久調書を作成するために、データを自動監査調書機構に持ちこむために本技法のログ機能が用いられている。

- ・自動量的分析（Automated Quantitative Analysis）

監査用のデータに対して一定の標準的な統計分析用のプロファイルを用意している。例えば最大、最小、平均（mean）および最頻値（mode）の統計、線型回帰分析のaおよびb要素（factor）などがある。

- ・自動質的分析（Automated Qualitative Analysis）

ユーザにエキスパート・システムの機能を提供するものである。質問はSASのマクロによって行われ、答は他のSASのデータベースに蓄えられる。全ての質問に対する答が用意された所で、分析用のプログラムが実行され、監査の対象となっている課題に対する意見をユーザに提供するものである。

② 継続監査法

イ. 目的と構成

本技法は、継続的に実施される監査業務の自動化を目的として、TSO CLIST、TSOのファイル、SAS言語によって書かれたプログラムにより構成されている。CLISTによって指定したSASのプログラムをIBM3084のバックグラウンド・プロセッサに投入し、どの本番用データ・ファイルを処理すべきかを判定する。SASプログラムは対象とするデータ・ファイルを自動的に読み込み、

そして種々のテスト、記録の選択、報告書の作成、種々のターミナルへのメッセージの提供などを実行する。

・CLISTの機能

CLISTによってどのSASプログラムを実行させるか判定する方法には2つの方法がある。

一つは、生成されたデータ・セット名を比較する方法である。前回生成されたデータ・セット名をリストする標準ファイルの名前が今回生成されたデータ・セット名と異なる場合に、初めて、そのファイルを対象とするSASプログラムが実行される。

もう一つの方法は、前回のSASの実行されたジョブ・ストリームのデータを利用するものである。前回特定のファイルが継続監査法の対象となってから、一定の期間が過ぎると、SASプログラムが自動的に実行されるものである。このジョブ・ストリームによる自動実行はCLIST上のマスタ監査ファイルによってコントロールされている。

CLISTが実行されるべきSASプログラムを決定すれば、CLISTはさらに実行用の適切なJCLと他の必要な技術的情報を作成する。完成した完全なジョブ・ストリームがここでバックグラウンド・プロセッサに投入される。

ロ. 機能

現在、継続監査法によって監査を実施しているのは、主として次の6つの業務に関するデータに対してである。

- 買掛金データ
- 請求データ
- 元帳データ
- 本番データ・ファイルへのアクセス・データ
- 本番用プログラムの変更のデータ
- 本番用プログラムの障害のデータ
- ・買掛金データ

買掛金の日次取引からランダムにサンプルが抽出され、事後監査のために別のデータ・セットとして保管される。また、特定の判断基準（金額、勘定コードなど）に合致する取引も全て選ばれ、別の

データ・セットとして保管され、証憑突合のために印字出力される。最後に、基本的な統計情報が累積されデータ・モデリングにおいて将来利用するために保管される。

- ・請求データおよび元帳データ

特定の判断基準に従って、請求データおよび元帳データは同様な方法でテストされる。抽出された取引は保管されると同時に、手作業によるフォローのために印字出力される。

- ・本番データ・ファイルへのアクセス・データ

MVSオペレーティング・システムによって作成されるSMF（IBM社のSystem Management Facilities）ファイルを用いて、特定のタイプのファイル・アクセスをテストするSASのジョブ・ストリームである。本社の会計、給与、買掛金のマスタ・ファイルの正規の本番用プログラムを通さない更新が行われた場合に、このようなファイル・アクセスのタイプを反映するSMFの記録の証拠を監視する。

- ・本番用プログラムの変更のデータ

この処理は、次に説明するプログラム変更コントロール法のインタフェースの役割をはたすものであり、本番用プログラムの変更を防止するために本番用ロード・モジュール・ライブラリ（PRODLIB）よりデータを抽出するジョブ・ストリームである。

- ・本番用プログラムの障害のデータ

このSASジョブ・ストリームは、SMFのデータを処理し、本番用プログラムの障害を示すアベンド・コード（abend code）をチェックする。最近の傾向を示した報告書を作成し、情報システム部の管理者にフォローアップのために提供される。

以上、ここに示したジョブ・ストリームが完了すると、それぞれ選択されたレコードの印字出力、ディスク・ファイルへの保管、それぞれの該当するファイルへの抽出データの件数、更新されたファイル名などのメッセージの送信が行われる。

出力報告書は、内部システム監査人にまず配布され、そして適切な手作業による監査手続が実施される。

③ プログラム変更コントロール法

イ. 目的と構成

本番用プログラムの変更が適切に行われていることを監査するために、自動プログラム変更統制のデータを保有する、PANVALET（ソース・コード）、PRODLIB（実行ロード・モジュール）、PROCLIB（JCLとジョブ実行用パラメータ）という3つの主要ファイルを対象として処理を行う。

ロ. 機能

監査の手順としては、まずPRODLIBまたはPROCLIBでの変更を識別し、次にそれらの変更をPANVALETのソース・コードの変更に突合し、さらに開発、保守の関連ドキュメンテーションを監査することになる。

・PRODLIBのデータに対する機能

PRODLIBは、標準的なロード・モジュール・フォーマットの分割型データセット（PDS: Partitioned Data Set）であり、このデータ・セットには個別メンバー名（本番プログラム実行ロード・モジュール）を記録した内部ディレクトリがある。内部ディレクトリには、各PDSメンバー名に対してひとつのデータ項目（フィールド）を設定し、そのメンバーのサイズを示している。

システム監査人は、メンバー名とサイズを読み、SASデータベースに保管するためSASプログラムを作っておく。毎日、同様のデータベースが作成され、前日のバージョンと比較される。そして、プログラムの変更、新規登録、削除があれば、そのメンバー名、サイズ、日付を他のSASデータベースに記録する。

また、このとき変更のあったプログラムについては、すべての実行ロード・モジュールが自動的に内部監査人のライブラリにコピーされ、そして、該当するソース・コードもPANVALETから、後でコピーされる。これらのコピーは次に説明する監査手続のために用いられる。

・CCR（Change Control Requests）との突合

A製材会社の情報システム部では、本番用プログラムの変更の承認を追跡できるように、オンラインでCCR用ファイルを管理している。このファイルと、既に作成されている一定期間の変更プロ

グラムのSASデータベースとを突合し、変更プログラム突合ファイルへ転記する。

週次に、突合ファイルからCCRの承認のない本番用プログラムの変更をリストした報告書を作成し、この報告書を情報システムの管理者と監査の責任者にフォローアップのために配布する。この過程で、次に述べる監査技法の対象となるプログラム変更が明らかにされる。

- ・PROCLIBのデータに対する機能

PROCLIBもPDSである。このデータでは、PRODLIBのようにフォーマット自身の変更を明らかにすることはできない。このため、PROCLIB上の変更を明らかにするために、各PDSメンバーごとに日々ハッシュ・トータルを計算し、前日のハッシュ・トータルと比較する方法がとられる。もし、JCLに変更が加えられれば新しいハッシュ・トータルが作成されることになる。

ハッシュ・トOTALの差が明らかになれば、そのメンバー名と変更の日付がSASデータベースに転記される。そして、新しいメンバーのコピーが監査ライブラリに登録される。

- ・ソースおよびオブジェクト・コンペア・プログラムの機能

PANVALETより抜き出されたソース・コードの変更およびJCLの変更は、定期的に内部監査人によって保持されている参照コピーと比較される。必要な場合には、クーパース・アンド・ライブランド(C&L)社製のソース・コンペア・プログラムを使って変更を明らかにし、該当するドキュメンテーションを証憑突合する。これらの変更プログラムの監査用ライブラリと参照コピーは、ソース・コードと実行ロード・モジュールの関連を検証するためにも用いられる。又、C&L社製のオブジェクト・コンペア・プログラムは、ソース・コードと実行ロード・モジュールとが一致していることを確かめる過程において利用される。

(4) 処理上の考慮事項

ここで紹介したコンピュータ支援監査技法を適用するためには、内部のシステム監査人によるバッチ・ジョブの実行が先ず必要である。監査データベース機構法の場合においても、次週以後の監査のスケジュールをもと

にSASデータベースが作成される。このデータベースは、週末の間にバッチ処理で作成され、月曜日の朝には利用できる状況にする必要がある。

継続監査法の処理の場合においても、毎日午前の早い時期にバッチ処理を走らせ、通常の監査人がフォローアップする時間を十分に取れるように配慮する必要がある。

プログラム変更コントロール法の場合には、PRODLIBより先ず抽出が行われなければならないため、継続監査法のジョブ・ストリームが終了した後に、バッチ・ジョブを走らせる必要がある。そして、その処理結果としてのデータベースがプログラム変更コントロール法の処理の中でさらにテストされていく。

このバッチ処理をいちいち指示しなければならないことが、まだ効率性を阻害している。この処理をより自動化することだけでなく、全体的なアプローチを考え直すことにより、より効率化をはかろうとしている。

(5) セキュリティおよびコントロール上の考慮事項

不当な本番用プログラムの変更を明らかにするために、コンピュータ支援監査技法を高度に利用していることから、本番用ロード・モジュールが変更されていないことを、独立的に判定するためにIBMのAMBLISTのユーティリティを利用している。しかしながら、このユーティリティは、実行に多大の時間を要するため（PRODLIBすべてをレビューするのに5時間程度を要する）、定期的に通常の監査手続が正常に実行されているのを確かめるための、クロス・チェックに利用されるにすぎない。

高度コンピュータ支援監査技法に対するセキュリティを維持するために、全ての監査用ソフトウェアとデータの管理は内部監査部で行っている。バックアップをとるだけでなく、常駐しているものは、ACF2（セキュリティ・パッケージ）を利用して保護している。

さらに、内部監査部では独立的にSMFファイルを対象としてSASプログラムを走らせ、監査用プログラムまたはデータに対して不当なアクセスが試みられていないかチェックしている。情報システム部は内部監査部のプログラムの保守と処理に関して一切責任を負わない。

(6) 高度コンピュータ支援監査技法の効果および問題点

イ. 効果

1982年に、当社では5名の専属のシステム監査人がいた。そして、

主としてデータ・センターのレビューとそのシステムに対する全般統制を監査していた。

高度コンピュータ支援監査技法を導入した結果、2名の専属のシステム監査人が専らこれらの監査を行い、他の3名は統合的な監査を一般の内部監査人と一緒に行い、監査の範囲を大幅に拡大した。

一般の内部監査人は、監査実施にあたって自由にデータにアクセスできるようになった。従来は、監査が始まってから、監査プログラムの処理結果が出るまで数週間待っていた。

プログラム変更およびデータ・ファイルへのアクセスの監査も短い時間で実行でき、監査対象となる母集団の範囲も大幅に増加した。現在利用している標準的コンピュータ・ルーチンも理解しやすく、保守も容易である。監査人は、よりリスクが高い領域に焦点をあてることができるようになり、異常点に対しても即時に対応できるようになってきた。

ロ. 問題点

高度コンピュータ支援監査技法の導入にあたっては、一般の内部監査人の反抗があるため、十分に実験を行い調査を実施する必要がある。このためにかかなりのコンピュータ資源を必要とする。当初異例の取引を監査人が入力し、そのフォローアップをユーザに質問するために、質問事項が多くなり、被監査部門は批判的であった。しかし、現在ではそのような状況になれてきた。

ハ. 全体として

従来に比べてシステム監査人への依存度が低くなり、一般の内部監査人が自由にそれぞれの目的にあった出力を作成できるようになってきたことである。システム監査人も含めて技術的スキルを向上させることができるようになってきた結果、監査に対する積極的な評価を行う傾向が強くなってきた。

(7) 将来のコンピュータ支援監査技法について

監査データベース機構法の標準レポートの機能を拡大する予定である。また、エキスパート・システムとしての量的分析の能力を向上させ、グラフィックと分析の機能を追加する予定である。

継続監査法については、ジョブ・ストリーム数をさらに増やす予定である。本番のデータベースに対するダイレクト・アクセスについても機能

の向上が必要である。

最後にメイン・フレームとマイクロコンピュータのより密な結合が研究されている。メイン・フレーム上で処理される監査プログラムとマイクロコンピュータ上の監査プログラムとの整合性をうまくとることを考えている。メイン・フレーム上の監査手続と処理結果をマイクロコンピュータにダウンロードし、監査調書を効率的に作成しようとするものである。

エキスパート・システムを使ったコンピュータ自動生成内部統制質問書の作成と、監査の必要なリスクの高い分野を把握することを容易にする、データ・モデルに基づいた予測テストを実施することも考えている。

3.1.3 B銀行

(1) システム監査の方針

内部監査部では、コンピュータ・システムの監査のために統合化アプローチを用いている。この方法では、データ・センターのインテグリティ・コントロールの監査結果とビジネス機能のアプリケーションの監査結果を統合させる。ビジネス機能の監査では、プログラム化された手続と手作業による手続が統合的に結合されて、コントロールの十分性、有効性、効率性が評価される。この監査は、一般の監査人とシステム監査人が共同して行う。このアプローチの長所としては、以下のようなものがある。

- 一般の監査人とシステム監査人の壁をなくする。
- 監査の有効性と効率性を増す。
- 経営管理者に対してリスクとその顕在化に対する全体的な評価を提供できる。
- 自動テスト技法の使用を増加させる。
- 内部監査部のコンピュータ・リテラシー (computer literacy) を高める。

(2) コンピュータ支援監査技法の必要性

銀行業界は、新しい商品、サービスの提供による厳しい競争環境のなかにある。このような新規商品およびサービスを情報システムが十分に支援する必要がある。このようなビジネスに内在するリスクおよびビジネスを支援する情報システムと、内部統制におけるコントロール上のリスクを効果的に評価するのが内部監査部の重要な役割である。

このような環境において、適時に実務的なビジネスの内容を評価し、コンサルタント／アドバイザーとしての役割を果たすことが内部監査人に期待されている。このような新規ビジネスの増加が、内部監査部がより効率的で、コスト・パフォーマンスをより重視した監査を実施することを強要している。

この結果として、監査ソフトウェアの利用が増加してきた。自動化されたリスク分析のツールを利用して監査資源の最適配置を決定し、効率性を高めるためにマイクロコンピュータを利用したフローチャートの作成、報告書作成、プロジェクト時間の管理、ワード・プロセッサなどである。

また、データ・センターやコミュニケーションに対する物理的、論理的アクセスやセキュリティを考慮する重要性も忘れてはならない。これらの評価にも監査ソフトウェアやセキュリティ・パッケージを利用することにより効率的に監査できる。

(3) 採用しているコンピュータ支援監査技法

B銀行で用いられているコンピュータ支援監査技法（CAAT）には、次のものがある。

- 専用監査ソフトウェア
- 汎用監査ソフトウェア
- ユーティリティ・プログラム
- プログラム比較プログラム
- テスト・データ法
- 汎用マイクロソフトウェア

これらは、データ・センターの監査と統合化された監査のために用いられている。これらを一覧表に示すと表3-1のようになる。

表3-1 B銀行で採用しているコンピュータ支援監査技法（CAAT）の一覧表

コンピュータ支援監査技法	ソフトウェア	納入業者	目 的
専用監査ソフトウェア	COBOL/ CULPRIT	IBM/ CULLINET	準拠性テストと実証性テストを支援する。
汎用監査ソフトウェア	EDP- AUDITOR	CULLINET	データ・センター業務を評価する。 統計的サンプリングの測定・評価をする。
ユーティリティ・プログラム	例えば IEHLIST	IBM/DEC	システム・ソフトウェアのデータを抽出する。
	ACF2	SKK	データのセキュリティを評価する。
	GURDIAN	ON-LINE SOFTWARE	データのセキュリティを評価する。
プログラム比較プログラム	SOURCE COMPARE	COOPERS & LYBRAND	アプリケーション・プログラムの 変更に対するコントロールを評価する。
テスト・データ法	—	—	プログラム手続の準拠性をテストする。
汎用マイクロソフトウェア	LOTUS 1-2-3	Lotus Development Corp.	準拠性テストと実証性テストを支援する。

多様な技法をプログラム化し、実行するのは、中堅または熟練した水準のシステム監査人によって行われる。ある特定のコンピュータ支援監査技法を用いるかについては、一般の監査人との協議の上決定される。

コンピュータ支援監査技法は、データ・センターの監査と統合された監

査の過程で利用される。コンピュータ支援監査技法が考慮される時には、上に示した技法の選択は現存する手作業の監査手続と比較され評価される。重要な評価基準は、監査の質・網羅性・効率性である。

(4) 専用監査ソフトウェア — COBOL, CULPRIT

COBOLとCULPRITで書かれたこのソフトウェアは、多様な監査目的を達成するために利用されている。不動産信託部門の最近の監査において、準拠性テストを支援するためにソフトウェアが開発された。全ての顧客レコードが、税識別番号を含んでいることを確かめるために、顧客マスタが読込まれた。また追加的に、全ての無利子のローンに対しては、ローン・ファイルと照合が行われた。これによって、監査効率が向上した。というのは、従来の手続は、監査人がプログラマにファイルのダンプ・リストを依頼し、これを手作業で調査するというものであった。不動産信託システムに大きな変更がなければ、このソフトウェアは、準拠性テストを支援するために毎年使用することができる。

専用監査ソフトウェアは実証性テストにも利用されている。顕著な二通りの利用法は、取引の再計算と確認状の作成である。例えば、試算表を再計算することにより、総勘定元帳の勘定残高を立証し、また、取引が適切に分類されていることを確認する手助けとなった。

この手続の代替手続は、貸借対照表上で取引が適切に分類されているかを、取引を手作業で集計することであった。

財務部門の監査でも、各種の金融資産の利子計算のプログラム化された手続をテストするために監査ソフトウェアが作成された。このソフトウェアがなくては、この利子計算を手作業によってテストすることは困難であったであろう。

要求払預金勘定の監査と同様に、各種の貸付部門の監査でも、確認状発行のために顧客残高を抽出するソフトウェアが作成された。

プログラム化された手続をテストするソフトウェアを作成するには、時間のかかる開発作業が必要になる。しかし、それを次年度以後も利用することにより、監査効率が向上してきた。

(5) 汎用監査ソフトウェア — EDP-AUDITOR

EDP-AUDITORパッケージは、数種類のルーチンから成る。監査の過程で、そのルーチン中の2つを選択して利用した。異常終了したジ

ジョブが適切に修正され再度実行されたか、また、承認されたバッチ・ジョブだけが実行されたかを検証するために、SMFルーチンを、また、統合化された監査を行うため、トランザクションの中からサンプルを無作為抽出するために統計サンプリング・ルーチンを利用した。

データ・センター監査の過程で、EDP-AUDITORのSMFルーチンを利用した結果、異常終了したジョブのSMFレコードのあるタイプが記録されていないことが検出された。また、承認されたジョブだけが実行されたことを検証する過程で、問題があった。というのは、ジョブの名称の付け方が標準化されていないため、テスト・ジョブと本番ジョブとを区分することが困難であった。従って、監査を実施した者は、SMFの出力リストからジョブ予定表に照合するかわりに、単にジョブ予定表からSMFの出力リストに照合することができただけであった。

結論として、監査チームは、次年度はEDP-AUDITORのSMFルーチンをコンピュータ・オペレーションの効率性の評価に利用すると決定した。

EDP-AUDITORの乱抽出ルーチンは、収益と費用の監査および貸借対照表の監査に適用された。収益と費用の監査の過程で、証拠資料と照合検証するためにEDP-AUDITORによって取引が乱抽出された。また、貸借対照表の監査では、勘定明細と証拠資料との一致を確認するために、勘定明細が乱抽出された。

(6) ユーティリティ・プログラム — DEC, IBM, ACF2, GUARDIAN

オペレーティング・システムの一部であるユーティリティとセキュリティ・ソフトウェア・パッケージは、広範に利用されている。DECやIBMコンピュータの環境下では、これらのソフトウェアは、主にオペレーティング・システムの情報を入手するためと、システム・ソフトウェア・ファイルを読込む目的で利用される。これにより、我々が理解しているオペレーティング・システムの環境が確認できるようになる。

ACF2やGUARDIANといったセキュリティ・ソフトウェア・ユーティリティは、データ・セキュリティのコントロールの評価に利用される。ACF2やGUARDIANを用いることにより、各種のログオン識別番号の属性、特定のアプリケーション・ファイルの保護規則、オンラインの取引に対するセキュリティ、システムに導入された各種のセキュリテ

ィ・オプションなどをリストした出力帳票が作成される。

(7) プログラム比較プログラム — SOURCE COMPARE

アプリケーション・プログラムに対する全ての修正が適切に承認されることを確かめるため、データ・センターの監査や統合化された監査の過程で、“SOURCE COMPARE”というプログラム比較プログラムが利用されている。前年のデータ・センターの監査の間にコピーされたソース・プログラムのライブラリと当年度のそののコピーとがソース比較プログラムによって突合比較される。プログラム・コードの各行は、前の版のものと比較され、異なっていれば、プログラムが変更されていると明確に表示される。従って、変更の承認が変更統制票（Change Control form）の上に記録されているはずである。

(8) テスト・データ法

テスト・データ法は、統合化された監査の過程で利用される追加的な技法である。監査部門が作成したデータを本番システムにインプットすることにより、テスト・データ法は複雑なコントロールをテストすることができる。全ての起りうる条件をテストするためには、代表的な取引のサンプルを寄せ集めることが大切である。この技法は、信託部門の監査でプログラムの手続を包括的にテストするために用いられている。

債券取引オンライン・システム（Bond Online Trading System）の監査の過程で、設計した通りに主要なオンライン処理が効率的に稼動しているという合理的な確証を得る手段として、テスト・データ法が選ばれた。

この技法を適用するに当っては、インテグリティ・コントロールの評価が不可欠であった。これは、プログラム化された手続（特にエディット手続）が監査を実施した範囲内で継続して運用されているという確信を得るために不可欠であり、これによりテスト・データの結果から引き出された結論が監査の期間を超えて遡及的に適用できる。

インテグリティ・コントロールは適当かつ効果的なものと判断され、テスト・データの作成が続けられた。監査の評価段階（妥当性）で識別された主要なエディット手続に対して、テスト・データが作成された。システムの予想結果がテスト・データと並行して作成された。予想結果は、テスト・データがインプットされ実際に作成された処理結果と比較された。

このテストの結果、債券取引オンライン・システムのオンライン・エディット機能が有効に働いているかについて、合理的な確証があるという結論に達した。十分で有効なインテグリティ・コントロールがあることから、監査の範囲と期間を超えて、この有効な機能が継続していることを確信した。

(9) 汎用マイクロソフトウェア

以下は、不動産信託部門の貸付機能の主要な局面をテストするために開発されたソフトウェアである。

このソフトウェアは、Lotus を利用した P.C ソフトウェアと、従来のコード化されたプログラムの両方を使用するように設計された。

① プログラムⅠ

このプログラムは、利子の再計算をするために Lotus を利用している。利子請求書は、再計算され A F S 試算表残高と比較された。全ての差異は最後の欄に書き込まれるようになっていた。しかし、何ら例外は検出されなかった。プログラム手続を再計算し、テストすることは、以前には行われていなかった。

② プログラムⅡ

このプログラム手続は、従来の監査では手作業で実行されていたが、いくつもの利子率があるため、その全てをテストすることはしていなかった。Lotus を用いることにより、全ての利子計算式を再計算した。その上、プログラム手続のテストを自動化したことにより、監査日数の約 2 日が節約された。

③ プログラムⅢ

このプログラムは、数種の準拠性テストと結合し、各項目の関係を評価する。例えば、プログラムⅣとⅤで以前に抽出した項目を利用するなどして、このプログラム手続のテストの結果、例外が発見され、監査報告コメントが出された。

④ プログラムⅣとⅤ

パーソナル・コンピュータによりポートフォリオ・レポート (portfolio position report) を利用する場合、ディスクのコピーを検索し、ドル金額順に並びかえる。このデータは、その後、他の監査ソフトウェアのインプットとなり、貸付書類のレビューの過程で利用される。以前は、

この作業は手作業で行われていたが、自動化により他の手続が楽になったと同様に2日の節約となった。

⑤ プログラムⅥ

このチェック・リストにより、貸付書類のレビューの効率が高まった。更に、様式の自動化によって、チェック・リストに記入された特定の項目についてのレポートや例外事項レポートが作成できるようになった。

⑥ プログラムⅦ

貸付価値率は、Lotus を使って再計算された。誤解が生じないように、注釈が付け加えられた。この手続は、以前手作業によって調書を作っていたのを自動化したものである。それが、たった1回だけ入力さえすれば、あとはボタンを押すだけでL/V（貸付価値率）欄が計算され、印刷される。

⑦ プログラムⅧ

この監査ソフトウェアは、監査でテストしたプログラム手続を再実行するものである。全ての取引は、システムで処理した日付よりも有効日付が新しいものだった。27日分の利子が回収されないと判明し、利子収入の損失が計算された。

(10) コンピュータ処理に関連したセキュリティとコントロール

最近、内部監査人は世界的規模で、監査部門の方針および手続のレビューを外部に委任した。というのは、これらの方針および手続はデータ・アクセスの能力、EDP部門のインテグリティ・コントロール、そしてコンピュータ機器およびソフトウェアの購入・リース・保管にまで関係しているからである。レビューの結果、これらの方針および手続は現時点では妥当ではあるが、今後更に次の手続を整備する必要がある。

- どのようにしてシステムやデータに対するアクセスが要求され、承認され、それがモニタされるかを規定する正式な手続が必要である。
- データ・センターのメイン・フレームのプログラム変更手続を概述する文書化された適切な手続が必要である。
- 各データ・センターで保管されている監査ソフトウェアやファイルの正式なバックアップとリカバリ手続が必要である。
- パーソナル・コンピュータ・プログラムやレポート作成プログラムおよびその他のソフトウェアの保守とライブラリ管理に関する文書化さ

れた手続が必要である。

- 部門のパーソナル・コンピュータのファイルとそのパーソナル・コンピュータの利用に対して正式なデータ・セキュリティ手続、特にメイン・フレームからファイルやプログラムのダウンロードをコントロールする手続が必要である。

(II) 将来のコンピュータ支援監査技法について

当部門では、将来利用される高度なコンピュータ支援監査技法の在り方を明確にしている。その一つは、オンライン・リアルタイムを基礎として、データやトランザクションを捕捉し、異常なものを識別し、その結果を分析する監査ソフトウェアの利用によって、継続的に監査が行えるようにすることである。現在認識しているもう一つの分野は、分析ソフトウェアであり、大きな監査リスクのある分野に監査資源を割当てるときに利用される。

銀行は、コンピュータ資源と通信ネットワークに大規模な投資をしているので、この分野の論理的アクセスに監査ソフトウェアを用いた監査が重要視されるだろう。

最後に、監査部門ではマイクロコンピュータを利用することにより効率化できるいくつかの分野を明らかにした。これらは、永久調書の磁気化、全ての監査調書や監査手続および監査報告書の自動作成などである。

3.1.4 C製薬会社

(1) 対 象

本事例では、病院に対するサプライを提供することを業務としている部門のみを対象とした。この部門は、11の米国内子会社を通して、販売、出荷、請求、回収業務を行っている（販売高10億ドル）。米国内物流拠点は、9～10カ所あり、7,000以上の顧客（病院など）を対象としている。取扱い品目数は数千以上あり、販売価格の条件設定も複雑なものとなっている。受注システムは、既存のパッケージを大幅に手直したもので、公衆回線（ローカル・オーダ・エントリ）、専用回線（リモート・オーダ・エントリ）およびコンピュータ間接続により入力が行われ、かつ、IMSデータベースが用いられている。

(2) システム監査の目標

取引量、システムの複雑度より判断し、内部統制、特に、外部からのデ

ータおよびシステムに関するコントロールが重要なものとなる。さらに、信頼あるシステムの開発、保守および運用のためのコントロールも重要な課題である。これらのコントロールが有効に機能していることを確認するための監査証拠の確保、災害対策、子会社間のインタフェース手続の確立なども考慮する必要がある。

(3) システム監査の方針

内部監査部では、次の諸点を考慮してシステム監査を実施している。

- システムとM I Sの両方の環境の変化に対応した特定のコントロール目標をマネジメントのために設定する。
- システムの開発過程において、コントロールが組込まれていることを判定するために、システム・ドキュメンテーションやM I Sの手続を評価する。
- コントロールを文書化する。
- システム・コントロール、手作業によるコントロール手続、全般的M I Sコントロールの改善案を作成する。
- 本番移行前に、M I S環境下においてインテグリティが確保されていることを確かめるためにテストする。
- テストの範囲、詳細テスト計画、テスト結果をレビューする。
- 子会社間におけるインタフェース手続をレビューする。

(4) 採用し得るコンピュータ支援監査技法

採用し得るコンピュータ支援監査技法には次のようなものがある。

- 総合テスト法 (I T F)
- テスト・デック／テスト・ケース法
- 監査ソフトウェア法
 - インテグリティ・コントロール監査用ソフトウェア
 - ・汎用パッケージ
 - ・個別仕様
 - アプリケーション監査用ソフトウェア
 - ・汎用パッケージ
 - ・個別仕様

(5) 総合テスト法 (I T F)

当技法は、以下の理由により採用しなかった。

- 。開発のための要員（人数・技術レベルとも）の手当が不可能である。
- 。システムに負荷をかける。
- 。システムが複雑化し、保守が難しくなる。
- 。特別なテスト・ケースを作成する必要がある。

(6) テスト・デック／テスト・ケース法

基本的には、開発過程におけるユーザによる詳細テストを監査の立場より利用しようとする方法をとった。詳細テスト計画は、システムのそれぞれ全ての機能要素をテストするために必要なテスト・ケースを各エンド・ユーザが作成したものであり、約百のユーザが数百のテスト・ケースを作成した。

監査人は、テスト計画が全てのビジネスの条件を盛り込んでいることを監査し、さらに、各テスト・ケースで実際に全てのプログラム化された手続をテストしているかを判定する。次に、ユーザのテスト手続を監査し、全てのテスト・ケースが実際に投入され、エラーが調査され適切に解消されていること、テスト結果が適切に文書化され承認されていることをレビューした。このレビューによって、監査人がテスト・ディスクを自ら作成することなく、このテスト・ケースで代用することができる。

詳細テストは、約6カ月に渡って実施され、この間に監査人は、継続的にその処理過程をレビューした。結果的に重要な問題はなく、本番移行後、現在までにおいても重要な問題は発生していない。

(7) インテグリティ・コントロール監査用ソフトウェア

これは、導入されているシステムの信頼性の度合を判定することを目的に、MISのオペレーション環境下におけるコントロールを検証するために必要となるソフトウェアである。インテグリティ・コントロールのソフトウェアを利用することなくこのような複雑な環境におけるコントロールをテストすることは不可能である。

当社で利用しているソフトウェアには次のようなものがある（⑥を除いてCoopers & Lybrand社製）。

① PARMLIBアナライザ

これは、IBM社のOSの機能を規定するPARMLIBから、次のような情報を抽出し、報告するものである。

- 。PARMLIBに規定された内容：IBM社の標準メンバー名とそ

の簡単な説明

- P A R M L I Bの詳細：各メンバーの詳細な説明とコントロール上の留意事項

② S L A P S

これは、プログラムの開発・保守およびセキュリティの監査に用いるものであり、リンク・ライブラリから次のような情報を抽出し、報告する。

- プログラムの変更：プログラム名、コンパイル日付、リンク・データなど
- Z A Pの使用状況：Z A P実行日、Z A P対照プログラム、Z A Pコメントなど
- プログラムの保守状況：前回と今回の監査の間に生じたプログラムの変更、追加、削除

③ ソース・コンペア

80バイトのソース・イメージで、修正前後の2つのプログラムの比較を行うものであり、追加・削除・修正が行われたレコードを識別することができる。比較結果は、旧プログラムおよび新プログラムの修正対象のレコードにフラグを付けたレポートとして出力される。

④ コア・イメージ・コンペア

同一プログラムのオブジェクト・モジュールとロード・モジュールの比較を行うプログラムである。これによって、本番に使われているプログラムがレビュー対象のプログラムと同一であることを調べることができる。

⑤ S M Fアナライザ

I B M社製のS M Fのデータを調べるためのプログラムであり、次のようなレポートを作成する。

- 異常終了したプログラム
- 指定日時に処理されたプログラム
- 指定データ・セットにアクセスしたプログラム

⑥ システム・ユーティリティの使用

A C F 2のルールを逆コンパイルしてセキュリティの構造とレベルを判定する。

(8) アプリケーション監査用ソフトウェア

① IMS データベース・アナライザ (C & L 製)

IBM 社製 IMS データベースに対して、適切なコントロールが組込まれているかを、

- a) 承認されたプログラムによってのみでデータにアクセスしているか
- b) 未承認のアクセスはパスワードによって防止されているか
- c) トランザクションやコマンドに対する端末への未承認アクセスが防止されているか

といった主要なコントロール・ポイントにより分析する。

② データ検索用ソフトウェア

このソフトウェアは必要でない。すなわち、システムの信頼性については別途テストし、信頼できるものと判定している。さらに監査用レポートの出力は、本番用アプリケーションに組込まれているからである。

(9) 処理上の留意事項

① 教育

C & L 社製のソフトウェアの使用法の習得のために、C & L 社のインストラクタのもとで2日間の研修が内部監査人に対して実施された。ソフトウェアが適切に利用されるように教育を実施することが先ず処理以前の問題として重要になる。

② 実行のタイミング

開発過程において実施する場合には、開発スケジュールを責任者と打合せて十分に把握しておく必要がある。また、データのインテグリティをテストする場合にもそのオペレーション・スケジュールに注意をする必要がある。

当初、内部監査人はタイミングの重要性をあまり認識していなかった。タイミングは、早すぎても遅すぎてもいけない。

③ 処理結果の文書化

十分に処理結果の意味を吟味し、文書化する必要がある。例えば、コア・イメージ・コンペアのプログラムの出力の読み方が、当初十分に理解できなかったため、その効果に疑問を持った。これは、C & L 社の研修を受けることによってその疑問点が解明された。

④ その他

これ以外に処理上の考慮すべき点としては、テストの判断基準（特に抽出サンプルの大きさ）、経営者との計画および十分なコミュニケーションの必要性、“凍結した”プログラム・ライブラリの安全なコピーの管理などがある。

(10) セキュリティ上の留意事項

監査用ソフトウェアをライブラリに常駐させるか否かについて検討した。検討の結果、次の理由により常駐させることにした。

- ACF2によるプロテクションが十分にかけられている。
- ライブラリから監査用プログラムを取出すのはそれほど高度な技術が必要としない。
- 監査人自身が本番用ソース・コードにアクセスするのはほとんど不可能である。

監査用プログラムへの不正アクセスは、SMFデータ、ACF2の分析によってモニタされている。現在までそのような不正アクセスの事実は発覚していない。

(11) 各技法の評価

① テスト・デック／テスト・ケース法

イ. 効果

実際に開発時に作成される詳細なテスト・ケースを監査人が独立した立場でレビューするので、重複作業が少なく、かつ、テストの目標と監査人の目標が一致しているため大変効率的なテストであるといえる。

また、オペレーション時においても情報システム部に余分な負荷をかける必要もなく、実際に用いられる本番システムをテストすることができるため効果的なテスト方法である。

これらのテストを十分に実施することによって、本番移行が問題なくスムーズに行われ、その後の監査においてもシステム上の重要な問題点が発覚していない。また、本番移行1カ月前に、システム全体にわたった十分な監査を完了させることができた。

ロ. 問題点

詳細なテスト・ケースが作成されるのを待って監査をスタートさせるため、タイミング上の問題が生じる。また詳細なテスト・ケースを

レビューするため、監査人と情報システム部の担当者との十分な調整、監査人自身の業務に対する知識が十分にあること、継続的に同スタッフに割り当てられることなど監査のスケジュール上負荷が大きい。さらに、テスト・ケース作成者の能力に依存する所が大きく、監査の質の均一性がとれないこと、監査調書にテスト結果を統合化する自動化された方法がとりにくいこと、特定のテスト・ケース作成のために時間がかかることなどが問題である。

② 監査ソフトウェア法

イ. 効果

監査ソフトウェア法を利用することによって、コントロールの準拠性テストを全くやめた。この方法を一度導入すれば、次回以降は迅速に、問題なく実行することができる。

この方法を適用することによって、技術的環境におけるコントロールの弱点を実証的に監査できる。そして、その結果に対する経営者による評価を、技術面を積極的に取り扱っていることにより、高めることができる。この方法を適用する他の効果としては、複雑なシステムの中における変化を迅速に明らかにできること、情報システム監査人の技術的能力の向上、将来の監査に使用するであろう当社環境の深い理解などを挙げることができる。

全体的にいうならば、従来の準拠性、実証性テストよりは、より効率的で効果的な方法であるということができる。

ロ. 問題点

この方法を実行するにあたって技術面における要求を満たしうる監査スタッフの教育が問題となる。監査ソフトウェアを実行し、その結果を解釈できなければこの方法の意味がない。この結果として、システムの詳細な知識を持つスタッフが継続的に同一業務に割り当てられ、監査担当者の割り当ての柔軟性がなくなる傾向がある。

また、実行にあたってはコンピュータ使用の時間が問題となる。情報システム部の本来の業務スケジュールを優先しながらも、本番システム移行日にできるだけ近い時点で実行し、その出力結果を評価し、フィードバックすることが必要である。

これ以外の問題点としては、あまりにも多くの出力があること、監

査ソフトウェアがセグメント化されているため、全体的な視点からのコントロール上の弱点／問題点を見過すことがあること、ACF2のレビューには時間がかかりすぎ、かつACF2のユーティリティの特殊な知識を必要とすること、一般の監査人との調整が十分に必要なことなどがある。

(12) 将来のコンピュータ支援監査技法について

現在の技法に関しては、以下のような改善を行う必要があると考えている。

- 手作業とコンピュータ化された手続のコントロール目標および判断基準を統合化した監査手続を作り上げ、一般監査人とシステム監査人の間の調整の必要性をなくし、お互にそれぞれの両者の間のわれ目に落ち込むリスクを回避すること。
- システム監査グループ内に技術支援機能確立し、スタッフの研修、ソフトウェアの保守、問題解決の支援を行うこと。このことにより、情報システム部への依存の減少、システム監査人の技術スキルのレベルの向上、継続的な特定業務へのスタッフの配置の減少が可能となる。
- 年1回の広範囲な監査を行うよりも継続的な監査の実施を行うアプローチを確立させること。このことにより、より頻繁にシステムをモニタできるだけでなく、監査担当者の割当ての柔軟性が増し、実際により多くの監査用ソフトウェアを実行させる環境における研修が、より多くのスタッフに対して可能となる。また、情報システム部の要員に対する影響を減少させることができる。

中期的な視点より、将来のコンピュータ支援監査技法の方向を考えると以下の点を考慮する必要がある。

- システム構造を分析し、テスト・ケースを生成するエキスパート・システムの開発により監査の時間を節減させる。
- ACF2については、よりユーザ・フレンドリーなユーティリティの利用により、特定のスタッフ、ユーティリティの依存度を減少させる。
- パッケージをより統合化されたものにして、監査人が全体的に始点より終点まで一環して監査できるユーザ・フレンドリーなものにする。
- 出力の量を減少させるため、異常点に焦点をあてた出力を考慮する。

3.2 国内における現状

3.2.1 調査の概要

(1) 調査目的

一昨年来、通商産業省による「システム監査基準」の公表や、システム監査技術者試験の実施開始など、このところシステム監査への関心が高まって来ている。しかし、システムの信頼性や安全性の監査を実施する際のツールや技法はまだ不十分であり、今後の充実が望まれる。このような状況に鑑み、システム監査支援ツールや技法を既に使用中、あるいは計画中と思われる先進企業を対象に、アンケート調査によって、現在のツール及び技法の問題点、新しいツールに対するニーズや意見等を聞き、今後開発すべきシステム監査支援ツールの要件に反映させることを目的とした。

(2) 調査方法

現状／ニーズ調査の方法は多様ではあるが、システム監査支援ツール・技法という高度に専門的なテーマに対しては、不特定多数に対して行うよりも、専門家の意見を聞く方法が適切と考え、システム監査に関して問題意識を持っていると思われる先進企業・会計／監査事務所・専門家を選択し、アンケート調査を行うことにした。

アンケートの基本方針としては、現状のシステム監査の実施状況（実施レベル、普及レベル）を勘案して、次の3つに分けて回答してもらうことにした。

- ① 既に、システム監査に支援ツールを使用している場合には、当該ツールに対するQ/A（A票）
- ② 現在、システム監査に支援ツールを使用していない場合、あるいは今後新たに使用したいツールがある場合には、そのツールに対するQ/A（B票）
- ③ システム監査支援ツールの使用の有無にかかわらず、情報処理システムの信頼性、安全性の監査のための支援ツール・技法についての意見または要望（C票）

(3) 調査内容

以下に示すような質問項目を設定した。

（アンケート用紙は付録を参照）

(ツールを使用している場合に回答)

〔質問 1〕 ツール・技法の名称

〔質問2〕 ツール・技法の種類

- | | |
|---------------------------------|--------------------|
| 1. 監査領域選択法 | 2. 得点法 |
| 3. 複数現場用監査ソフトウェア | 4. 汎用監査ソフトウェア法 |
| 5. トランザクション選択法 | 6. 組込み監査データ収集法 |
| 7. テストデータ法 | 8. 基本事例システム評価法 |
| 9. 総合テスト法 | 10. 平行シミュレーション法 |
| 11. スナップショット法 | 12. マッピング法 |
| 13. トレース法 | 14. コード比較法 |
| 15. 統制流れ図法 | 16. 監査手引法 |
| 17. ジョブ会計データ分析法 | 18. 災害テスト法 |
| 19. システム開発時点統制ガイドライン法 | 20. システム開発ライフサイクル法 |
| 21. システム検収及び統制のグループ | 22. 導入後監査法 |
| 23. カバレッジ・テスト | 24. オンライン・モニタ |
| 25. その他〔 〕 | |

当該ツール・技法が、1～24のいずれの種類に該当するかを見るためである。1～24のツール・技法はSACレポート（日本語版）の分類に準拠した。（アンケートの際には、SACレポートのツール・技法についての説明から抜粋して添付した。）

〔質問3〕 ツール・技法の概要

1. 区 分〔a. 購入, b. 自作, c. 改造〕
2. 使用形態〔a. 会話形, b. バッチ形〕
3. 被監査システム〔a. オンライン, b. バッチ〕
4. 機 能 概 要
5. 入 力 情 報
6. 出 力 情 報
7. そ の 他

「区分」は、（a）当該ツールを市販品として購入したのか、（b）監査のために自社で新たに作成したのか、あるいは（c）既にあるプログラムを改造したか、を見るためである。これにより、監査支援ツールの商品性を判断することができる。

「使用形態」は、（a）当該ツールが会話形（オンライン）で使われるのか、または（b）バッチ形で使われるか、を見るためである。これにより、システム監査支援ツールのオンライン化の状況がわかる。

「被監査システム」については対象とするシステムが、（a）オンライン・システムであるのか、あるいは（b）バッチ・システムであるのか、を見るためである。これにより、システム監査の対象としてオンライン・システムがどの程度取り上げられているかの傾向がわかる。

「機能概要」、「入力情報」、「出力情報」、「その他」は、自由記述である。これらは、当該ツールの機能全般を把握するためであり、新しいシステム監査支援ツールの要求仕様（機能要件）を検討する上での参考となる。

〔質問4〕 監査対象

- | | |
|--------------|--------------------------------|
| 1. データ・ファイル | 2. システム（OS、DBMS等） |
| 3. プログラム（AP） | 4. 手続き・方針・標準 |
| 5. 人（要員） | 6. その他〔 〕 |

当該ツールがどんな対象、すなわち、データを監査するのに使われるのか、あるいはプログラムを監査するのに使われているのか等を見るための質問である。これにより、システム監査支援ツールを使う対象が把握できる。

〔質問5〕 使用段階

- | | | |
|---------|---------|---------|
| 1. 企画段階 | 2. 開発段階 | 3. 運用段階 |
|---------|---------|---------|

使用段階は、当該ツールがシステム開発ライフサイクル（SDLC：System Development Life Cycle）のいずれの段階で使われているかを見るための質問である。これにより、現在のツールがどの段階に、より多く使われているかが把握できる。

〔質問6〕 監査目的とツールの有効度合

- | | |
|------------------|------------------|
| 1. 準拠性の監査〔上、中、下〕 | 2. 信頼性の監査〔上、中、下〕 |
| 3. 安全性の監査〔上、中、下〕 | 4. 効率性の監査〔上、中、下〕 |
| 5. 有効性の監査〔上、中、下〕 | 6. 採算性の監査〔上、中、下〕 |
| 7. その他（ | ）〔上、中、下〕 |

当該ツールが情報処理システムのどの側面（準拠性、……、採算性）を監査するのに使用されているかを見るための質問である。これにより、現在のツールが特にシステムの信頼性あるいは安全性の監査にどの程度用いられているかが把握できる。

〔質問7〕 ツール・技法の使用効果

- | |
|---|
| 1. 監査作業の効率〔a. 大変良い, b. 良い, c. 変わらない〕 |
| 2. 監査結果の精度〔a. 大変良い, b. 良い, c. 変わらない, d. 良くない〕 |
| 3. 監査の範囲〔a. 広がった, b. 変わらない〕 |
| 4. 監査期間〔a. 短縮した, b. 変わらない, c. 長くなった〕 |
| 5. その他 |

ここでは、当該ツールを「監査作業の効率面」、「監査結果の精度面」、「監査の範囲面」、「監査期間面」という側面で自由に記述してもらう。これにより、そのツールの有効性の程度が把握でき、新しいシステム監査支援ツールの機能要件を検討する上での参考となる。

〔質問8〕 ツール・技法の問題点

- | |
|---------------------------|
| 1. 適用限界 |
| 2. ハードウェア環境整備 |
| 3. ソフトウェア環境整備 |
| 4. 使うための準備（インプット・データの準備等） |
| 5. 操作の容易さ |
| 6. アウトプットの情報項目、質等 |
| 7. 被監査プロジェクトまたはシステムへの影響 |
| 8. その他（改善点、要望点等） |

上記の側面で問題点を自由に記述してもらう。これによって、現在のツールにどのような改善点や問題点があるかが見極められ、新しい

〔質問3〕 監査対象

- | | |
|---------------|--------------------------------|
| 1. データ・ファイル | 2. システム (OS, DBMS) |
| 3. プログラム (AP) | 4. 手続き・方針・標準 |
| 5. 人 (要員) | 6. その他〔 〕 |

ツール・技法を使用して今後監査したい対象を見るための質問である。内容はA票と同様である。

〔質問4〕 使用段階

- | | | |
|---------|---------|---------|
| 1. 企画段階 | 2. 開発段階 | 3. 運用段階 |
|---------|---------|---------|

ツール・技法を使用して今後SDLCのどの段階を監査したいかを見るための質問である。内容はA票と同様である。

C 票

(ツール使用の有無にかかわらず回答)

情報処理システムの信頼性、安全性の監査のための支援ツール・技法についての意見または要望。
--

システム監査支援ツール・技法に対する意見や要望などを自由に記述してもらい、新しいシステム監査支援ツールの要件を決める際の参考とする。

(4) 調査対象

システム監査の対象企業は全産業分野にわたるが、日本でのシステム監査の実施レベルを勘案して不特定多数の企業ではなく、システム監査に対し先進的と思われる企業のみを対象とし、31社を選択した。

表3-2に、その業種別内訳を示す。

表 3 - 2 業種別調査対象の内訳

金 融 業	6 社
生 命 保 険 業	2 社
商 社	3 社
石油製品製造業	3 社
その他製造業	10 社
運 輸 業	2 社
監 査 法 人	2 社
情報処理サービス業	2 社
電 力 事 業	1 社
合 計	31 社

アンケートは郵送方式をとり、付録に示す「システム監査支援ツール・技法の実態調査票」を各社1セット（A票3部、B票1部、C票1部）ずつ送付し、記入を依頼した。A票の3部は、複数個のツールに対応するためである。

本調査は、昭和61年7月より8月にかけて行った。

3.2.2 調査結果の評価・分析

(1) 回答状況

本アンケート調査は31社に対して行い、22社からの回答を得た（表3 -

表 3 - 3 業種別回答状況

	回答数	回答率
金 融 業	5 社	83 %
生 命 保 険 業	1 社	50 %
商 社	2 社	67 %
石油製品製造業	2 社	67 %
その他製造業	6 社	60 %
運 輸 業	2 社	100 %
監 査 法 人	1 社	50 %
情報処理サービス業	2 社	100 %
電 力 事 業	1 社	100 %
合 計	22 社	約 71 % (平均)

3 参照)。回答率は約71%で、この種のアナケート調査に比べ回答率が高く、調査対象に選定された企業のシステム監査に対する姿勢、関心の高さを如実に表している。

A票に対しては14社、30票の回答があった。その内訳を表3-4に示す。概して銀行関係から多くの回答が得られた。

表3-4 A票に対する回答状況

回答票数	件数
6票	1社
4票	1社
3票	1社
2票	6社
1票	5社
合 計	14社

B票に対しては11社(約35%)が回答をよせ、その内、A票には回答せずにB票に対してのみ回答している企業は9社(約29%)である。調査対象に選定した31社は、我国でのシステム監査の先進企業であるが、回答企業22社中9社(約41%)はまだ実際にツールを使用していないということがわかり、日本でのシステム監査の実施レベルの低さが実感された。

C票に対しては20社(約65%)から回答があり、現在のシステム監査支援ツールに対し多様な問題点の指摘が得られた。

(2) 結果の評価・分析

① A票に対する評価・分析

A票の集計結果の要約を表3-5に示す。

表 3-5 A 票の集計結果の要約 (その 1) (回答数: 14 社)

I. ツール・技法の名称	II. ツール・技法の種類										III. 区分・処理形態	IV. 監査対象					V. 使用段階			VI. 監査目的と有効度合							VII. 使用効果				備考		
	4汎用監査ソフトウェア法	5トランザクション選択法	7テストデータ法	9総合テスト法 (ITF)	10平行シミュレーション法	13トレース法	14コード比較表	17ジョブ会計データ分析法	19システム開発時点統制方法	20システム開発ライフサイクル法		nその他	1データ・ファイル	2システム (OS/DBMS)	3プログラム (AP)	4手続き・方法・標準	5人 (要員)	6その他	1企画段階	2開発段階	3運用段階	1正確性	2信頼性	3安全性	4効率性	5有効性	6採算性	7その他	1監査作業の効率	2監査結果の精度		3監査の範囲	4監査期間
AUDEX-100	○											a	b	b	○						○	◎	◎						◎	◎	◎	◎	
AUDEX-100	○											a	b	b	○		△				○	○	○						○	○	◎	◎	
EASYTRIEVE	○*				○	○						a	a	b	○						○	△	◎	◎	△	△	△		◎	◎	◎	◎	*: 4GL
EASYTRIEVE	○*				○							a	b	b	○		△				○	○	◎						○	○	◎	◎	
EASYTRIEVE (IBM用) QLP (ユニパック用)	○*	○										a	b	a・b	○					○	◎	◎	◎						◎	◎			
EASYAUDIT (or PANAUDIT)	○											a	b	b	○						○		○	○					○	○	◎	◎	
EDP-AUDITOR (CULPRIT)	○											a	b	a・b	○						○		◎	○					○	○	◎	△	
F O C U S	○*											a	a	b	○					○	○	○	○						○	○	◎	◎	
SELECTZ と CONPRN (取引抽出及び確認状 作成システム)	○	△										a	b	b	○						○	○	◎	○	△	△	△		○	○	○	◎	
CAAP (Computer Assisted Audit Programs)	○	△						△		△		b	b	a・b	○	○					○	◎	◎	△	○	○	○		○	◎	◎	○	EASYTRIEVE と FOCUS の流用
汎用監査ソフトウェア法 (各種ソフトウェアの利用) AUDITPAK II (ACOS-6用)	○	△			△							c	a	b	○						○	○	◎	◎	○	○	△		◎	◎	◎	◎	

(注) 記号の意味 (IV欄 ◎: 上, ○: 中, △: 下), (VII欄 ◎: a, ○: b, △: c, x: d)

表 3-5 A 票の集計結果の要約 (その 2)

I. ツール・技法の名称	II. ツール・技法の種類										III. 区分・形態	IV. 監査対象					V. 使用段階	VI. 監査目的と有効度合							VII. 使用効果				備考							
	4 汎用監査ソフトウェア法	5 オンライン・システム選択法	7 テストデータ法	9 総合テスト法 (ITF)	10 平行シミュレーション法	13 トレース法	14 コード比較表	17 ジョブ会計データ分析法	19 システム開発時点統制方法	20 システム開発ライフサイクル法		n その他	1 区分 (a:市販品・b:自作・c:改造)	2 使用形態 (a:オンライン・b:バッチ形)	3 被監査システム (a:オンライン・b:バッチ)	1 データ・ファイル		2 システム (OS/DBMS)	3 プログラム (AP)	4 手続き・方法・標準	5 人 (要員)	6 その他	1 企画段階	2 開発段階	3 運用段階	1 正確性	2 信頼性	3 安全性		4 効率性	5 有効性	6 採算性	7 その他	1 監査作業の効率	2 監査結果の精度	3 監査の範囲
テストデータ法			○									b	a・b	a・b	○		○						○		○						○	○	◎	○		
テストデータ法			○									b	b	a・b	○		○						○	○	◎	○	△	○	△			○	○	○	○	
テストデータ法			○									b	a・b	b			○	○					○	○	○	◎	◎				◎	◎	◎	◎		
テストデータ法			○									b	a	b			○						○		◎						○	○	◎	◎		
テストデータ利用による 監査技法			○	○									a・b	a・b			○					○	○	○	◎	○	○	△	○			○	◎	◎	○	
①開発用 (オンライン・システム) ②研修システム			○	○								上井用 開発用	a・b	a・b			○						○	◎	◎	○	△	△	△		◎	◎	◎	◎		
テストデータ法 (テスト及び教育機能)			○	○								b	a	a			○	○				○	○	◎	◎	◎	○	○			○					
総合テスト法				○								b	a	a			○	○					○	○	◎	◎	◎				◎	◎	◎	◎		
K-AUDIT (普通預金監査システム)					○							b	b	a			○						○	○	◎	◎					△	○	◎	○		
パソコンによる平行テスト					○							b	b	b			○						○	◎	○	△	△	△	△		◎	○	○	◎		

(注) 記号の意味 (IV欄 ◎: 上, ○: 中, △: 下), (VII欄 ◎: a, ○: b, △: c, x: d)

表 3-5 A 票の集計結果の要約 (その 3)

I. ツール・技法の名称	II. ツール・技法の種類										III. 区分・形態	IV. 監査対象					V. 使用段階			VI. 監査目的と有効度合							VII. 使用効果				備考				
	4 汎用監査ソフトウェア法	5 トランザクション選択法	7 テストデータ法	9 総合テスト法 (ITF)	10 平行シミュレーション法	13 トレース法	14 コード比較表	17 ジョブ会計データ分析法	19 システム開発時点統制方法	20 システム開発ライフサイクル法		1 区分 (a:市販品・b:自作・c:改造)	2 使用形態 (a:合納形・b:バッチ形)	3 被監査システム (a:オンライン・b:バッチ)	1 データ・ファイル	2 システム (OS/DBMS)	3 プログラム (AP)	4 手続き・方法・標準	5 人 (要員)	6 その他	1 企画段階	2 開発段階	3 運用段階	1 正確性	2 信頼性	3 安全性	4 効率性	5 有効性	6 採算性	7 その他		1 監査作業の効率	2 監査結果の精度	3 監査の範囲	4 監査期間
プログラム 照合プログラム							○				b	b	a		○							○	◎	△	◎				◎	◎	◎	◎			
COMPARE X							○				a	b	a・b		○						○		◎	◎						○	○	○	○		
コード比較法 (IBMのユーティリティ)							○				a	b	b		○							○			○										
売掛金監査用ソフトウェア										ソフトウェア	b	b	a		○	○		不正防止				○	○		○						○	○	◎	◎	
検査資料作成 システム											b	b	a				○	○	不正防止 常時性			○	◎		◎					◎	◎		◎		
SAFE (Security Administration Feature)										アプリケーション				○	○						○	○		○	◎					○	○	◎	◎	日立のプロダクト	
システム 事前 監査								○			b	a・b	a・b		○	○					○		○	○	○					◎	×	◎	◎		
P R I D E									○		c						○			○	○		◎	○	△	△	◎	◎						また協力を実施していないので効果は不明	
「標準化」,「システム管理」 担当 係 の 設 置										21	b	a・b			○	○	○					○	○	◎	◎	○	◎				◎	◎	◎	◎	

(注) 記号の意味 (IV欄 ◎:上, ○:中, △:下), (VII欄 ◎:a, ○:b, △:c, ×:d)

イ. ツール・技法の種類

最も使用者が多いのは汎用監査ソフトウェアであり、11社（2種類以上使用している企業を除くと9社）が採用している。実際に使用されている汎用監査ソフトウェアは、AUDEX-100、EASY-AUDIT（PANAUDIT）、EDP-AUDITOR（CULPRIT）、SELECTZ & CONPRN、CAAP、AUDITPAK II の6種類である。その他、分類を同じにしている、EASYTRIEVE、QLP、FOCUSは、汎用監査ソフトウェアというよりは第4世代言語あるいはデータベース問合せ言語であるが、提供機能及び使い勝手の面からシステム監査支援ツールとして流用されたものであろう。汎用監査ソフトウェアは、また監査対象のトランザクションを選択するためにも使用されている（4社回答）。

2番目に多いテストデータ法は7社が採用している。テストデータ法は技法であり、専用ツールがあるわけではないが、システム開発時に使用されるテストデータ・ジェネレータを流用することは可能である。

3番目に多い総合テスト法（ITF）は4社で採用されており、その内3社は銀行業である。ただし、これら銀行3行は、研修店、講習店等の方法を用いており、厳密には総合テスト法ではない。残りの1社は製造業で、総合テスト法を定義通りに実施している。

次にコード比較法は3社で採用されており、その内の2社は銀行業で、残りの1社は情報処理サービス業である。2社は市販のツール（COMPAREX、IBMユーティリティ）を利用し、1社は自作のツールを使用している。

次の平行シミュレーション法は2社で採用されており、その内の1社は情報処理サービス業、残りの1社は銀行業である。両者とも、普通預金の利息計算に対して適用している。

ロ. 区分と処理形態

汎用監査ソフトウェア法とコード比較法は大半が市販品を使用しているが、その他のツール・技法は自作品を利用している企業が多い。

使用形態及び被監査システムは、会話形またはオンラインとバッチ

が半々である。被監査システムがオンライン・システムの場合、オンラインで監査すべきではあるが、使用するツール（主として、狭義の汎用監査ソフトウェア）がオンライン利用をサポートしていないため、バッチ方式の監査を行っている場合がある。

ハ. 監査対象

監査対象としては、データ・ファイル（14件）、アプリケーション・プログラム（18件）、手続き・標準等（8件）などが大勢を占め、OS等のシステム（2件）や要員（2件）などは少ない。特に、汎用監査ソフトウェア系統のツールはほとんどデータ・ファイル対象であり、一方、テストデータ系統のツールはほとんどアプリケーション・プログラム対象である。

ニ. 使用段階

使用段階は、運用段階が圧倒的に多く、一部開発段階にも使用されている。

現在のシステム監査の実施レベルは、運用段階のデータ・ファイルに対するシステム監査であると言えよう。

ホ. 監査目的とツールの有効度合

監査目的として、大半が「準拠性」及び「信頼性」の監査と回答しており、次に「安全性」である。現在のシステム監査支援ツールは、「効率性」、「有効性」、「採算性」などの監査の支援には不適當であるという評価がうかがえる。

ヘ. ツール・技法の使用効果

システム監査支援ツールの効果は、「監査作業の効率化」、「監査結果の精度向上」、「監査の範囲拡大」、「監査期間の短縮」の順にその目的達成に寄与しているようである。

② B票に対する評価・分析

B票の集計結果の要約を表3-6に示す。

イ. ツール・技法をシステム監査に使用していない理由

B票に回答した11社の大半は、「システム監査の方法を構築している段階である」（6社）、「システム監査の体制が整っていない」（5社）、「システム監査の必要性をやっと認識した段階である」

表 3 - 6 B 票の集計結果の要約 (回答数：11社)

〔 () 内の数字が回答件数〕

I. ツール・技法をシステム監査に使用していない理由(回答有りのみ、複数回答)

- (6) システム監査の方法を構築している段階である。
- (5) システム監査の体制が整っていない。
- (2) システム監査の必要性をやっと認識した段階である。
- (1) ツール・技法が使いづらい。

II. 使ってみたいツール・技法 (回答数が2社以上のみ、複数回答)

- (4) テストデータ法
- (4) 汎用監査ソフトウェア法
- (3) 監査領域選択法
- (3) 総合テスト法
- (3) システム開発時点統制ガイドライン法
- (3) 組込み監査データ収集法
- (3) 監査手引法
- (3) システム開発ライフサイクル法
- (3) 導入後監査法
- (2) 災害テスト法
- (2) 統制流れ図法
- (2) 基本事例システム評価法
- (2) トランザクション選択法

III. ツール・技法を使用するときの監査対象 (回答有りのみ、複数回答)

- (7) データ・ファイル
- (6) プログラム (AP)
- (6) システム (OS, DBMS等)
- (4) 手続き・方針・標準

IV. ツール・技法の使用予定の段階 (複数回答)

- (9) 運用段階
- (7) 開発段階
- (2) 企画段階

(2社)という状況であり、システム監査支援ツールを使ってどのようにシステム監査をやるかの段階にはまだ達していない。

日本では、まだシステム監査をどのようにして普及させるかという段階ではあるが、安価で有効なシステム監査支援ツールがあれば、普及が促進されよう。

ロ. 使って見たいツール・技法

殆んどのツール・技法を一様に「使って見たい」と回答しており、システム監査の目的・対象に対する確固たる方針がまだ十分検討されていない様子が反映されている。

ハ. ツール・技法を使用するときの監査対象

監査対象としては、「データ・ファイル」、「プログラム」、「システム」ともほぼ等しく(7～6社)回答しており、この3つを重点的かつ同程度に監査したいと考えていることを表わしている。

ニ. ツール・技法の使用予定の段階

使用予定段階では、「運用段階」が圧倒的に多く(11社中9社)、次が「開発段階」(11社中7社)であり、「企画段階」は少ない(11社中2社)。

システム監査は、「運用段階」、「開発段階」、そして「企画段階」というシステム開発ライフサイクル(SDLC)の実体とは逆順に実施されることを反映しているように思われる。システム監査も管理の一種であり、管理サイクル(PDCA)と同様に、まず企画段階から実施するのが正しいと言うべきであるかも知れないが、監査のし易さは確かに逆順であろう。

③ C票に対する評価・分析

C票に対する意見を以下に整理・分類して示す。

イ. 汎用監査ソフトウェアに対する意見

- 最近出回りつつあるが、DBへもアクセス可能で安価な汎用監査ソフトウェアが欲しい。
- 従来の汎用監査ソフトウェアは機種が限定されているものが多い(例:IBM系以外は使用できない。小型機では使用できない)。メーカー、機種に制約なしに使用できる汎用監査ソフトウェアが欲しい。

- OSの如何にかかわらず、汎用的に使える監査プログラムが欲しい。
- 当行においては、「汎用監査プログラム」を使用しておらず、監査対象に合ったソフトウェア（パソコン・ソフトを含めて）、ツールを使用して監査を実施している。
- 最近、特に「汎用監査プログラム」の使用が叫ばれているが、各社ごとにシステム、プログラム、ドキュメントが違うので、「汎用的」な監査プログラムで、より精度の高い監査が可能なのか、若干、疑問を感じている。
- オンライン・システムに対しては、内部監査の目的の設定にもよるが、一般的な監査支援ツールのもつ統計機能、標本選択、合計照合等の機能は、使用効果に疑問がある（公認会計士の検査は別）。ただし、各種監査支援ツールの考え方は、システムに組込む監査機能に参考になる。

ロ. 簡易言語に対する意見

- 各監査によってある程度の変化がつけられ、かつ容易にプログラムを作成できる簡易言語を使用した現方式（CAAP方式：Computer Assisted Audit Programs）は、当社の監査に適した方式だと思われる。従って、簡易言語の機能の向上が望まれる。
- 現在は、DBについてはDML（Data Manipulation Language）を使用し、オフコン・レベルには簡易言語パッケージを利用して、主としてデータ選択によって必要なデータを抽出している。
- 従来、EDPシステム利用監査の一環として、監査対象の選定を効果的に行うために、DBよりユーザ言語を用いて抽出する手法を採用してきた。

ハ. 組込み監査データ収集法に対する意見

- 当社は、昭和50年頃、社内で組込み監査データ収集法（監査モジュール法）に相当するシステムを開発し、5年間程実際に利用してみた。しかし、監査対象がそれぞれ取引形態が異なり、抽出条件をその都度修正したりはしたが、もう一つ効果が上がらなかった。但

し、これは監査対象の取引の異常性を抽出するシステムとしては有効であり、他にも使えると思っている。

ニ. 監査証跡に対する意見

- 。 通常、システム監査の段階では、キーになるファイルには必要な監査証跡があるかをチェックしているので、以後はCAAPで監査を行うのが容易になる。

ホ. 監査ガイドラインに対する意見

- 。 当社には、ICEG (Internal Control Evaluation Guide) - New York にて開発したガイドラインがあるが、ガイドラインそのものはまだコンピューターに入っていない。

ヘ. アクセス制御プログラムに対する意見

- 。 現在使用中のアクセス制御プログラム (SAFE: Security Administration Feature /日立製) では、アクセス規則の作り方に難しい点がある。
- 。 当社では、セキュリティ・パッケージ (現在、"SECURE" から "ACF2" に移行中) を使用して一般的なセキュリティ制御を行っているので、これらのパッケージが提供している監査用のレポートを利用することができる。

ト. 他の監査支援ツールの使用に対する意見

- 。 入力時に編集を行うデータについては、監査時に同機能の維持照合プログラムをかけて、ZAPの介入を監査している。

チ. 開発用ツールの流用に対する意見

- 。 多くのツール・技法には、システム開発部門における開発過程での処理技法、あるいはシステム・テスト段階 (単体テスト、システム・テスト) でのテスト技法に類すると思われるものがある。
- 。 支援ツールは監査人用に特化するのもよいが、むしろプログラマ用ツールを監査人が使いこなす方式の方が安上がりで、かつ実際のであろう。

リ. パソコン活用に対する意見

- 。 支援ツールが大型機用パッケージの場合に、パソコンにて事前にフォーマット・チェックや文法チェックを実施出来るものもある

(例：IBM/PCによるAudexのチェック)。大変良いことである。

ヌ、新しい監査対象分野に対する意見

- O A化、分散化の進展により、ミニコン、オフコンの使用が多機種に渡り導入されているが、これらに適用できる監査支援ツールが無い。問題である。
- 情報ネットワーク化や広範囲な分散システム化に対応しうる監査技法が統合的に考えられなければならない。従来技法の他に、新たにコミュニケーション・ネットワークならびに分散システム、分散データベース(DBMS)の監査技法/ツールをも加えなければならないと考えている。

ル、監査エキスパート・システムに対する意見

- 監査のためのエキスパート・システムの出現を待望している。

ヲ、オンライン・システムの監査機能に対する意見

- 監査機能に対する考え方：オンライン・システムが主流となっている現状においては、信頼性、安全性の監査機能についても即時性が要求されるのは当然と考えられる。特にセキュリティ対策という面からの監査機能は、オンライン・システムに組込まれていることが必須であり、データの受付段階からチェックしていく必要がある。

○ オンライン・システムの監査機能：オンライン・システムにおいては、次の様な検証機能を持たせ、運用のその場でチェックする仕組みが重要と考える。

- ① オンラインのチェック機能の充実。
- ② 重要取引については再チェック機能を持たせる。
- ③ トータル照合機能を持つ。
- ④ ファイル間の整合性チェック機能を持つ。

なお、これらの運用場面でのチェックから漏れるもの、または現場の統制状況等を検証のために必要なものは、検査用データを抽出後検証することも必要となる。以上の様な監査機能が体系的、効果的に仕組まれていることがオンライン・システムとして必要と考える。

- 結論：オンライン・システムにおける監査機能は、システムの一

部として組み込みを考えるべきものであり、一般的にいう監査支援ツールの機能は当然必要であればシステムに組み込まれていなければならない。システム監査は、これらの機能組み込みの状況の検証とその実行状況の検証が重要と考えられる。

ワ. 監査支援ツールの有効性に対する意見

- 。 支援ツール・技法は、あくまでも手段であって、目的ではない。従って、その時点における監査の目的に沿った支援ツール・技法を採用して（場合によっては組合わせて）行うことが大切である。
- 。 ツール・技法の有効性は評価しているが、システム監査、イコール、ツール・技法との発想ではなく、当面は周辺監査を中心に、徐々に監査テーマ・目的、監査手続に合致するツール・技法を採用していく考えである。
- 。 現在、システム監査の支援ツール・技法として一般的にあげられるものは、システム監査における一局面、一部分を個別にサポートするものであり、全体をカバーする体系化されたものではない。システム監査人が、これらの支援ツール・技法を駆使し、有効なシステムの信頼性、安全性の監査を実施するためには、これらを統合し、体系化された統合監査技法として集大成することが必要と考える。
- 。 この監査に当って最も専門的なツール・技法を必要とするのは、データ・ファイル、システム、プログラムを対象とする場合であると考えられる。従って、これら技法について「具体的に何が出来るか」、「何が検証出来るか」を、ユーザとしてはっきり認識しなければならない。
- 。 情報処理システムの信頼性、安全性は、システムの開発段階におけるシステム開発計画、システム機能テスト及びその検討の良否によって大きな差違が生じる。また、システム監査を効果的（経済的、時間的）に実施できるのもこの開発段階であるので、企画・開発段階で使用できる監査支援ツール・技法の研究を進めたいと思います。

カ. 監査体制／内部統制に対する意見

- 。 最近、システム監査への取組みについての経営方針が明確にされ、システム監査に対する具体的方向付けを検討、さらに人材育成を行っている段階である。

- 情報システムの監査を実施するためには、内部統制の整備と適切な運用がその前提である。内部統制が整備され、意図した通りに機能しているかどうかを検証することがシステム監査であると考えられる。従って、ツール・技法もこの点を考慮して開発する必要がある。
- 当社のシステムの現状（オンライン・リアルタイム・システム）に照した監査テーマ・目的、監査手続を検討した段階で必要なツール・技法を選択する所存ではあるが、現時点では具体的にツール・技法を特定するまでには至っていない。
- 今後、検討、開発して行かねばならぬ分野と考えているが、十分な検討はしていない現状のため、特に意見はない。

ヨ. ツール・技法のチュートリアルに対する意見

- 現状、この面の照会、調査が観念的で、支援ツール・技法の並列的な比較、解説が多いように見受けられる。もっと具体的、実践的に特定の監査目的ごとに、その目的に沿った監査アプローチ、あるいは技法・ツールの紹介が行われるようになると、EDPシステム監査が一般の人にとっても身近なものになると思われる。

例：特定のA業務システムの信頼性を監査したい。……

どのようなアプローチを行い、どのようなツール・技法を使えばよいのか。

- 監査目的に沿ったツール・技法が的確に選び出せるように、これらのデモンストレーションを希望する。
- 現在、ツール・技法についての文献はアウトラインが中心であり、具体的適用とその成果についての意見を聞ける機会が多くありませんが、今後、導入を検討するにあたり、この種の意見交換の場を設けてほしい。
- ようやく、いろいろな監査パッケージが米国から導入されてきているが、使いこなすのが一苦勞である。これから報告書を取りまとめる際に、実例、適用例、手法に重点を置かれない。
- 本調査票には、多数のツール・技法（24種）が添付されているが、現在市販されている参考文献で具体的に理解でき、監査実務に実際に利用するイメージが湧くものは少ない。これらのツール・技法の

実際の活用例を具体的に掲載した参考書及びツール・技法のリスト等の出版が待たれる。

- 24種類のツール・技法を使用して監査を行った事例を教えてほしい。
- 支援ツール・技法についての実務セミナーを地方でも開催してほしい。
- 某社のコンピュータに利用できる汎用監査ソフトウェア・パッケージの種類と価格を教えてほしい。

4. 情報システム開発技法とシステム監査



4. 情報システム開発技法とシステム監査

4.1 情報システム開発の動向

コンピュータを用いた情報システムを設計、開発するための技法は、これまで多数のものが考案され使用されてきた。システムの分析から設計、開発の一連の技法を手順化して、マニュアル化したものは、既に1960年代の初期頃から作られていた。IBM社のSOPなどはその一つである。

それらの技法を技術的課題で分類すれば、ハードウェアとしてのコンピュータの能力を最大限に引き出すためのもの、巧妙なアプリケーション・システムを構築するためのもの、保守性の高いソフトウェアを作るためのもの、あるいはデータベースやネットワークを構築するためのものなど多種多様である。

これらの多くは、企業におけるコンピュータ化の歴史の中で、時代、時代の要請に応じながら、システム開発に携わる者に対する標準的手順に組み込まれて提供された。

当初、少数の専門家を対象としたものも、やがて、大量の需要が見込まれたソフトウェア開発要員を確保する使命を持ち、さらに、システムの一貫性や安全性を追及するためのものとなってきた。このように時代の変遷と共に、その期待あるいは課題は変化している。

システム開発技法の変化を理解するためには、それを必要とする情報システムそのものの変遷を理解する必要がある。

4.1.1 情報システムの発展段階とシステム化課題の変遷

情報システムが、一つの企業に構築され発展して行く過程で、その特性がどのように変化するかを示したものに、有名なR. L. ノーランの6段階説がある。

ノーランは、第一段階を『創始期』と呼び、熱狂的な革命家が登場しコンピュータ化の橋頭堡を築く時期とした。第二段階を『拡大期』と呼び、革命が成功して、一挙にコンピュータ化領域の拡大が図られる時期としている。第三段階は、第二段階で繰り返された個別システム開発の結果、もたらされるシステム劣化を是正するための統制が組織化される時期とし、『統制期』と呼んでいる。

システム化の目的もステージを追うごとに変化する。つまり、第一段階は

コンピュータ化の利点を認めさせることを主眼に、目に見えやすい人手削減などの直接効果の実現やユーザニーズのきめ細かい把握と対応に重点がおかれる。第二段階は、短期間にコンピュータ領域を拡大するために、拙速を旨とする開発効率が求められ、ソフトウェア生産性向上、あるいはソフトウェア開発要員の大量動員を可能ならしめる施策に重点がおかれる。

第三段階では、第二段階で生じる歪みを是正するため、システム間の整合性を確立すること、あるいはシステム保守性を確保すること、さらにシステム安全性の確保などを狙いとして、システムの企画、開発、保守を統制することが重点課題となる。

今日、システム監査やシステムにおける内部統制に関する論議が盛んになったのも、多くのコンピュータ・ユーザが既に、第三段階にシフトしたからとも考えられる。それはデータベースやネットワークを過半数のユーザが既に導入済みであることを見てもうなづける。なぜならば、本格的なデータベースやネットワークはシステムの整合性や一貫性を確保する統制機能なしには構築しえないからである。

システム監査とシステム開発方法論の在り方を探るとき、第一段階や第二段階でとられてきた、これまでの方法論を前提とするのではなく、第三段階にふさわしい方法論を明らかにした上で、システム監査との関係を論じるべきと思われる。

4.1.2 拡大期における方法論

70年代後半から80年代の初頭、多くのコンピュータ・ユーザは、ノーランのいう第二段階の真っ只中にいたと考えられる。この間、一つの企業で開発されたプログラムの本数をみても、数%から数十%の勢いで成長している(図4-1)。

また、システム開発プロジェクトの期間も、長くて2年、多くは半年から1年と言う期間で一つのシステムを建設している。

これらの要請に応えるべくシステム開発方法論、あるいは技法にも次のような改善と強化が図られた。

(1) プロジェクト管理技法の強化

主として工程管理を重視し、進捗の把握と評価のための方法が強化され、次のような技法が考案された。

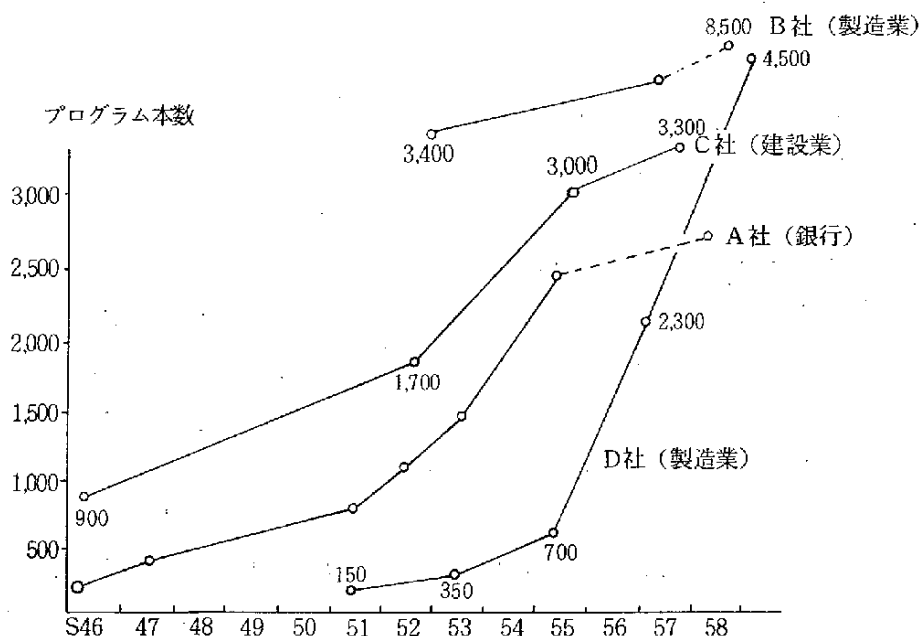


図 4 - 1 システム成長の状況

- ・開発工程の標準化
- ・フェーズド・アプローチ
- ・作業用ドキュメントの標準化
- ・プログラム・テスト技法とプログラム完成度評価技法

(2) 要求定義技法

プロジェクトの納期遅延や予算超過の主たる原因は、開発途中でのシステム仕様の変更にあるという前提から、早期に仕様を確定する方法が強化され、次のような具体策が講じられた。

- ・要求記述技法の標準化
- ・ニーズ把握と分析のための手順の標準化
- ・分析、計画フェーズへの時間配分の強化
- ・要求変更の統制

(3) プログラミングの標準化

ソフトウェア開発の手工業からの脱却あるいは工業化を合言葉に、主と

してプログラミング作業の標準化が進められた。次のような技法が普及した。

- 構造化プログラミング
- 構造化プログラム設計
- 構造化ワークスルー

4.1.3 拡大期の方法論がもたらしたもの

拡大期に、ソフトウェア・エンジニアリングの名の下に提案された技法の多くは、本来、ソフトウェア信頼性の向上、あるいは開発後の保守作業の容易さを目的にするもので、場合によっては、開発効率を下げてでも品質と保守性を確保させようとするものであった。

しかし、これらの技法は、必ずしもその意図通りに適用されたとは限らない。それらを導入したユーザのなかには、短期開発を目的に開発効率確保を期待したものも多い。それらのユーザは、一連の構造化技法を、単に新人教育用の手段として用いたか、標準化のみを最大の目的として用いたかの、いずれかであったといっても過言ではない。構造化技法によって開発効率の飛躍的向上が期待できないことを知ったユーザは、何等かの形で定着しかけた標準化を前提にプログラム開発支援ツールなどの機械化を進め、より大量のソフトウェア・コードを生み出す仕掛けを築くことになった。

拡大期における開発技法は押しなべて、プロジェクトごとに実施される個別システム開発の効率化を目的に編成され利用された。つまり、システム開発の問題点は、すべてソフトウェア生産性の低さにある、という前提で単位時間当たりソフトウェア・コード生成量を増やす方法に集中していた。

プロジェクト単位に開発効率を求めるシステム開発はコンピュータ化領域の短期拡大という点では、それなりの成果をおさめたといえるが、一方で次のような脆弱性をシステムにもたらした。

- 無体形な個別システムの累積
- 運用管理組織の管理能力を凌駕するソフトウェア規模
- データなど本来共有されるべき資源の重複
- 予期せぬプログラム間結合の増大と複雑性の増加

これらの脆弱性は、具体的な現象として、システム保守費用の急増、一貫性欠如、安全性欠如、膨大なバックログ、などをもたらした。これらの現象

これは、ノーランが第一段階、第二段階の特徴として指摘しているように、「統制の不在」下での成長の結果、もたらされる一種の劣化現象と見なせるものである。特に、情報システム全体に関するインテグリティ・コントロールへの配慮が二の次にされた結果といえる。

4.2 拡大期方法論の欠陥と統制期における方法論

4.2.1 拡大期末期の情報システム・イメージ

第二段階の末期から第三段階の初期に、企業の基幹業務のシステム化が不整合を内在しながらも取りあえず一巡する。そのようなユーザは、コンピュータ技術的には高度なレベルに達しており、オンライン・システムもデータベースも導入済みである。複数台のコンピュータが非集中または分散形態で稼働している。ソフトウェア開発支援ツールも活用され、何等かの規則や標準が制定されている。エンドユーザには端末やワークステーションが手渡され、限定された範囲ではあるがデータベースの活用が進められている。また、OA化の旗印の下で、パソコンやワープロ、あるいはファクシミリの導入が積極的に進められている。

しかしながら、一方で、膨大なソフトウェアの維持のために多大な労力と費用の支出を強いられている。度重なる保守で劣化したソフトウェアがソフトウェア資産の名の下に管理されているが、管理元は業務対応に設置されている開発グループである。システム変更要求は担当グループが窓口となり個別に調整され、担当者の判断でプログラムの変更が実施されている。各開発グループは例外なく膨大なバックログを抱えている。また、やむを得ずシステム変更を受付、実施した時はその影響がどこに及ぶかを特定できず、実施後のトラブルに振り回される。

データベース管理担当者も置かれているが、DBMS 廻りの技術的作業の負担を負担するだけでデータ標準の管理は行われない。総じて、企画、開発以外は運用業務と見なされ、コンピュータ、データベース、ネットワークなどは一切、運用グループに任される。しかし、それらを資源と見なした管理形態とはとれない。業務の単調さを嫌って優秀な人間は運用業務を回避し、それがために運用は外部に委託される。

以上が、システムの形態や規模だけは第三段階に達しながらも、システム

の開発や運用に関する形態は依然として第二段階のままであるユーザの典型的イメージである。しかし、今日、大勢を占めているのはこのようなユーザといえる。

4.2.2 拡大期方法論の欠陥

第二段階の拡大期によってもたらされる様々な歪み現象や脆弱性は、その方法論が共通的に持つ、次のような欠陥に基づくものといえる。

(1) システム設計制約の不備、曖昧性

ソフトウェアを主体とする情報システム設計がもつ最大の弱点は、設計制約の曖昧性にある。ハードウェア・システムなど、ソフトウェア以外のシステムではその設計対象物を規定する物理法則が存在する。ソフトウェアは質量もなく、その構造を規定する物理法則も存在しない。したがって、設計作業を規定する強引な公理化を図らねばならない。構造化技法はソフトウェア構造に関する公理化を、規則や因果律によって行った。そのため主としてプログラマの思考や作業を規定するような方法がとられた。このことが設計作業を難しくしただけでなく、設計結果のテストや評価をも難しくし、システムの実質的な吟味を困難なものにしている。

(2) コントロール・ガイドライン不在

納期重視の開発形態で、過度な開発効率の追及は、本来、組み込まれるべき不可欠なコントロールさえ迂回させるものとなる。さらに問題となるのは、設計技法として、コンピュータ処理性能の向上策やユーザ要求の分析方法などが強調され、コントロールに関する技法は体系立てて教育されていないことである。システムの信頼性に関する意識は高いが、信頼性技法もコンピュータ技法の一環としてとらえられる。したがって、バックアップやリカバリに関する技法は重視されるが、それがシステム全体のインテグリティを維持するコントロール技術の一部であるという認識は低い。

コントロールに対する認識の低さは、次のような開発形態や開発方法にも顕著に表れている。

① 内部牽制機構の不在又は結合

- ・ 同一人によるデータとプログラムの設計
- ・ 同一人による設計と開発の実施

- ・同一人による開発とテスト
- ・データなどの資源管理機能の欠如
- ・プログラム処理の結合

②プロジェクト又はサブシステムごとの資源設計と手配

- ・業務ごとのデータベース設計

③可視的証跡の迂回

- ・監査証跡の迂回

④業務処理の過度なプログラム化

- ・人手介入の極度な排除

⑤データベースへのデータの過度な集中

(3) ユーザ・ニーズ主導のプロジェクト万能主義

拡大期は、短期間にコンピュータ化領域を拡大することが重要な課題となり、特定課題を対象としたプロジェクトが編成されシステム開発が進められることは既に述べた。

プロジェクトは本来、有限な期間をもつ一過的な組織である。したがって、その権限の範囲はそのプロジェクトが存在する期間だけであり、また、そのプロジェクト内に限定される。さらに、プロジェクトの評価は、与えられた課題を予定の費用と予定の期間でいかに達成するかによって決まる。したがって、他プロジェクトに対する配慮や、長期的、間接的な課題への配慮は行いがたい特性をもつ。さらに、プロジェクトは目的志向性が強い組織である。その目的達成のためには、自らの作業の進めかた、方法論を選択する権限をも持つ。したがって、プロジェクトによるシステム開発は次のような弊害を生みやすい。

- ・孤立した一貫性のないシステム群
- ・システム・ライフサイクルへの配慮の欠如
- ・全社的構想や計画に対する整合性の欠如
- ・共通のシステム基盤の立ち遅れ

(4) 変わり得る要求の強引な固定

プロジェクト方式の開発の一つの宿命は、プロジェクト進行中に目的や要求仕様が変化すると、多大なやり直しを余儀なくされ、莫大な費用の浪

費となることである。したがって、プロジェクトの開始前、あるいはその初期の過程でシステム要求仕様を確定し、揺るぎないものにする方策がとられる。要求定義工学（リクウィアメント・エンジニアリング）とよばれる技法が要求の早期確定を目的に登場した。システム要求には事前に確定出来るものと、そうでないものがある。確定できないものを強引に決定させ、仕様凍結と称して事後の変更を一切受付ない方法は、結果的には、稼働後のシステム変更を多発させるだけでなく、プロジェクトの開発作業そのものの論理的・一貫性をも損うものとなる。

(5) ソフトウェア開発プロセス重視の密室開発

効率を追及するシステム開発は、勢い熟練した専門家集団による作業推進を求める。特に、煩雑なプログラム作りはコンピュータ部門内のプログラマの手に委ねられる。また、プログラム開發生産性を向上させる方策の一つとして、よく採られる方法は、プログラマを外部の雑音から遮断することである。ユーザや業務との拘り合いを敢えて断ち切ることでさえ行われかねない。したがって、プログラマは、自己の設計を、与えられた仕様に基づきさえすれば、比較的自由的な制約下で行うことができる。プログラムとして処理の正当性と性能さえ確保すれば、業務からの制約、あるいは保守性、拡張性などの制約は従属的なものとなる。

4.3 内部統制を前提としたシステム開発方法論

4.3.1 システム開発、運用における統制のあり方

今日、システム監査が論じられるとき、直接、監査ポイントや監査ガイドラインに関心が集中し過ぎる傾向が強い。

本来、監査とは『内部統制を評価し改善勧告を行う活動』といえる。そこには監査の前に内部統制が存在するという前提がある。したがって、まず、情報システムにおける内部統制に関する体系とその構成原理に関する論議が行われるべきと思われる。

(1) 内部統制の構成原理

統制（control）とは、目標や計画を達成するために必要となる制御活動を意味する。伝統的会計分野では内部統制（internal control）と

と呼び、1949年、A I A が定義した概念がある。即ち、『企業内部において、その資産を保全し、会計資料の正確性と信頼性とを検証し、さらに経営能率を増進し、かつ定められた管理上の諸方策の推進に役立てられる組織計画、及びあらゆる調整方法と諸手段とを含む』というものである。

さらに、伝統的会計統制には、統制の原則として次のようなものが定義されている。

- ・両立し得ない職務の分離 (segrigation)
- ・担当の限定と責任の明確化 (isolation)
- ・監督の強化 (supervision)
- ・内部監査 (internal auditing)

（2）統制の原則と統制方式

これらの原則は、経験的なものではあるが、感情と意志をもった人間が介入するという事実を踏まえたものである。システム開発も同様に人間による行為であり、その誤謬や改ざんは当該企業の資産や信用に多大な損失をあたえる。したがって、システムの開発、保守、運用の総ての局面、あるいは単一のシステムだけではなくシステム全体について、上記のような相互牽制を主旨とする内部統制を埋め込むための原理、原則が定義されなければならない。

（3）統制の原則と統制方式

（2） 効率的な統制方式の追及

今後、情報システムは、好むと好まざるとに拘わらず不特定多数のエンドユーザとの直接的拘わり合いをもつエンドユーザ主導型システム開発、あるいはエンドユーザ直接利用形態と呼ばれるものへ移行せざるを得ない。したがって、システム開発や保守の統制、あるいは運用や利用の統制も、不特定多数を対象としたものにならざるを得ない。システムに組み込まれる統制についても同様である。専門家でない不特定多数を対象に経済的、効率的な統制を実現する方策が追及されなければ、統制過度となり警察国家のごとき暗黒社会を構築することになりかねない。統制実現方式についても再考が必要となる。

（4）統制の原則と統制方式

①規則、ルールによる統制

統制を実現する方式の一つは、規則やルールを設けその準拠性を監視す

るものである。その方式の欠点の一つは、規則の認識、理解をその適用者に求めることにある。また絶えずその準拠性を監視し、違反を摘発できなければ有効性を失う。特にシステム開発やソフトウェア開発を規則で統制することは、その規則やルールを理解すること自体が特定の能力を要求する。したがって、特定少数を対象とした統制とならざるを得ない。

②仕掛けによる統制

望まれるもう一つの方式は、規則やルールによる統制を、仕組みや機構の不備によるものと見なし、むしろ規則やルールを排除しようとするものである。つまり、被統制者が統制されていること自体を意識しないで済むような統制の実現を狙う。例えば、『両立し得ない職務の分離』や『担当者の限定』などの原則を、規則によらず、むしろ『両立できない仕掛け』、『専任の仕掛け』をもって実現しようとするものである。そのための一つのアプローチは、システムの開発、運用担当者や利用者が、最も拘わり合う諸資源を統制することであろう。

第二段階におけるシステムのスラム化は、開発プロジェクトによる資源自給自足が最大の原因ともいえる。

1970年代の中頃からデータベースが急速に普及した。その目的はデータ資源の重複を排除することにあった。しかし、当初データベースを統制手段とみる者は少数であった。データベース化は必然的に、データ設計タスクとデータ利用タスク、プログラミング・タスクの分離を実現する。したがって、資源の自給自足を抑制し統制の基盤を築くものとなる。今日のデータベース・ユーザの多くは、意識しない内に、不特定多数を対象とした統制の仕掛けを築いてきたといえる。

データに限らず、コンピュータやネットワーク、ワークステーションも共有資源であり、ソフトウェアの一部も同様である。したがって、今後のシステムにおける統制は資源管理を前提に構築されるものとなろう。

4.3.2 システム開発、運用における統制と監査のフレームワーク

監査とは、内部統制を評価し改善勧告を行う活動である。また、その評価と勧告を行う者は独立な第三者でなければならない。似たようなものに、品質保証 (quality assurance) や検査 (inspection) がある。いずれも

システムの健全性、正確性などを目的とするものであることに変わりはない。しかし、その立脚点は異なるといえよう。

品質保証や検査は直接、対象とするシステムやソフトウェアを評価、吟味するのに比して、監査は、あくまでも、内部統制を評価するものである。したがって、直接、対象システムを評価、吟味して保証を与えるものではない。もし、監査を品質保証や検査と同等のものと見なすと、監査人自身が直接的な統制活動を負担することになり、統制制度や機構そのものの評価、吟味の立場を失うことになる。なんでも良いから、評価や吟味が行われればよしとする考え方も、統制の在り方、その機構、制度を探る観点からは危険なものといえる。

システム開発、運用における統制ならびに監査のフレームワークを図4-2に示す。即ち、システム及びシステム化活動（開発、運用）を統制するためには、まず、統制機構として直接、システムやシステム化活動を制御する機構が存在しなければならない。その機構は恒常的に存在し連続的な役割を果たす。一方、監査機構は統制機構を評価、制御するための機能を果たす。統制機構と監査機構が統一して機能を果たすような統制原理、原則が存在する。これら三つの機構は、それぞれ独立であることが求められる。

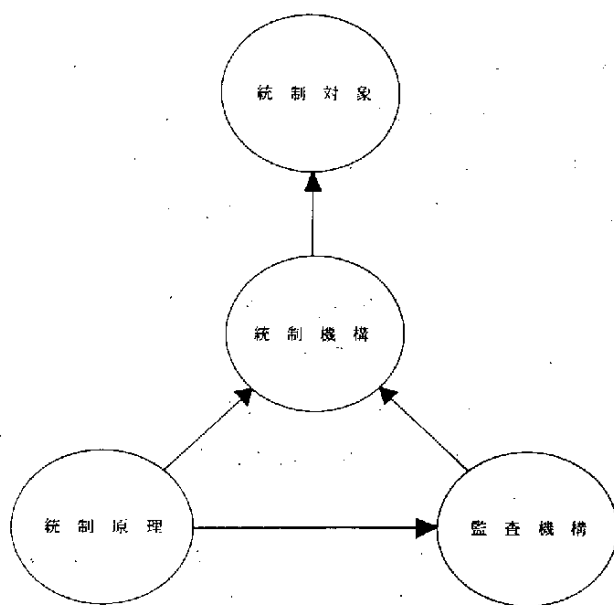


図4-2 統制・監査のフレームワーク

これらの機構はさらに、図4-3のように展開される。

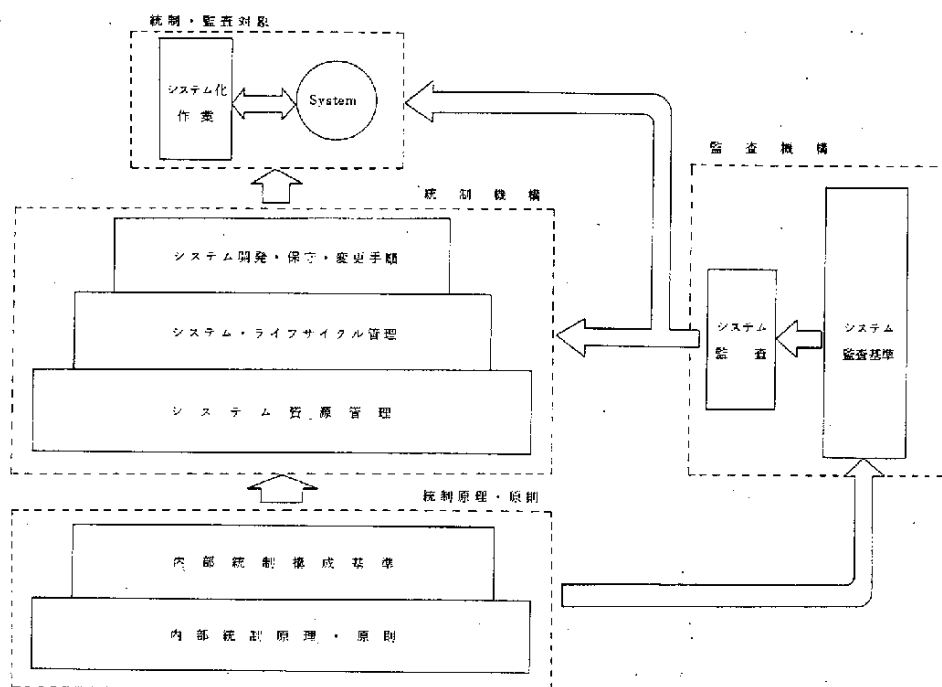


図4-3 内部統制フレームワーク

(1) 統制対象

一切のシステム化活動及びシステム。システム化活動は、開発、保守、変更及び廃止を包含する。また、システムそのものに対する統制とは、そのシステムが物理的に具備する統制メカニズムや統制手段を指す。

(2) 統制機構

次のようなコントロール手段から構成される。

① システム開発、保守、変更手順

システム化活動を構成する一切の作業を規定する手順。手順そのものは統制ではないが、統制を実現する手段となる。

② システム・ライフサイクル管理

システムの発生から消滅に至るライフサイクルに介入して、システム化活動の起動、評価、停止を制御する役割。

③ システム資源管理

コンピュータ、ネットワーク、さらにデータなどの資源を集中管理し統制の基盤を提供、維持する役割。

(3) 統制原理および原則

情報システムの開発、運用に関する統制全般に関して、その構成原理や基準を明示したもの。情報システムにおける内部統制をどのように構成すべきかの原理、原則を定義したもので、開発手順や運用手順を規定する基礎となるものである。

一般に、手順や規則にその目的や目標概念を明示することは難しい。概念的理解よりも実用性が求められるからである。開発手順や基準を行政における省令にたとえるならば、原理、原則は法律や憲法に該当するもので確固たる法理念をもって立法化されるものである。理念なき規則ほど有害なものはない。

(4) 監査機構

統制機構を統制原理、原則にてらして評価、吟味する役割で、統制原理、原則に合致する監査基準をもつ。監査機構は不定期に、統制機構の不備や妥当性を検証し、改善勧告を行えばよい。監査機構の役割は個々のシステム開発のすべてに直接関与することは不可能である。その役割はシステム・ライフサイクル管理のものである。個々のシステム開発の検証はシステム・ライフサイクル管理に依拠した上で、むしろ、情報システムに及ぶリスクを分析して統制機構全体の要件設定を行う責任をもつものと考えべきであろう。

したがって、システム監査の役割は、監査機構とシステム・ライフサイクル管理の二つが分担するものと考えられることもできる。

4.4 システム・ライフサイクルとシステム監査

システムの開発とその稼動、運用を一貫して管理統制する考え方は、システム・ライフサイクル管理として、1970年代の中頃から認識されだした。特に、ソフトウェア・エンジニアリングの観点からは、ソフトウェア・コストを開発コストだけで見るのではなく、保守コストを含めたライフサイクル・コストで計るべきだとする考え方が主流となった。その背景には、一つのソフトウェアのライフサイクル・コストの内、60から70パーセントが保守コストで占められるという事実があった。したがって、ソフトウェア・エンジニアリングの主

目標は保守コストの削減に置かれ、保守性に重点を置いた構造化技法が一般化した。

4.4.1 システム・ライフサイクルとその統制

(1) システム・ライフサイクルとは

システム・ライフサイクル (system life cycle) とは、システムがたどる、その生涯 (system life) に存在する過程のすべてを言う。勿論、その過程をどのように認識するかは立場によって異なるが、どのシステムも発生→稼動→消滅という過程をもつことは共通している (図4-4)。

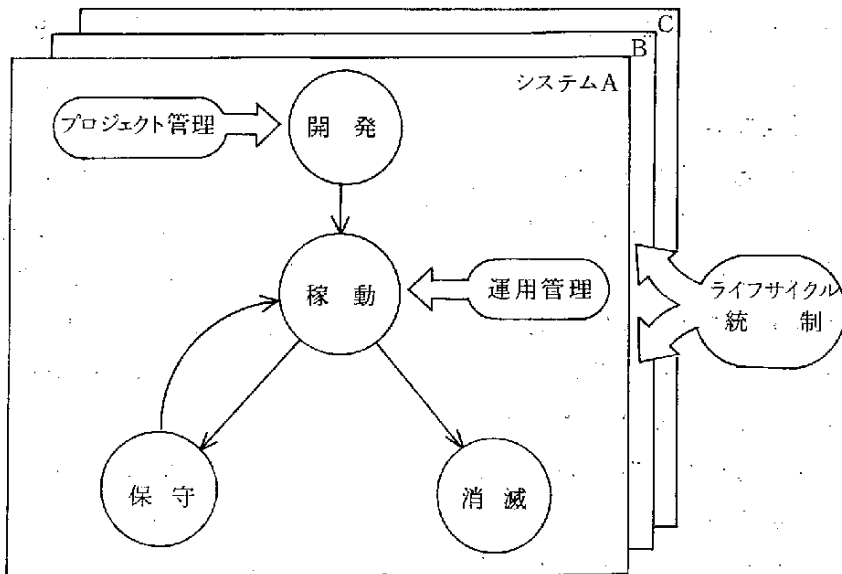


図4-4 ライフサイクル

システム・ライフサイクルの認識は、本来、システムの寿命に対する認識から生じたものと言えよう。つまり、その狙いはシステム又はソフトウェアの寿命を監視し、寿命の尽きたものを適切に消滅させることにある。それはまた、存在の価値がないものを誕生させないことにも通じる。

一般に、企業の業務を支援するシステムのソフトウェアは、数年という寿命を持つものである。その間、そのソフトウェアが開発という状態にいるのは数箇月に過ぎず、残りは稼動状態に置かれる。しかし、そのソフトウェアが管理されるのは開発状態にある数箇月だけである。運用管理も単

に、そのソフトウェアの稼動に関するものだけで、寿命に関する管理は行われない。このことが無統制なシステム変更、保守を許してきたといえる。

システム・ライフサイクル管理の目的は、短期的な開発よりも長期にわたる稼動状態を管理することにある。したがって、システム・ライフサイクル過程の認識も、開発過程のみを重視した分割（図4-5）から、稼動過程を重視した分割（図4-6）が必要となろう。

(2) システム・ライフサイクル統制の要件

システム・ライフサイクルの統制は、システム・ライフサイクル管理機能がこれにあたる。先に述べたようにシステム・ライフサイクル管理はシステム監査人と共に、システム監査活動を行うものといえる。

開発重視のこれまでの体制から、システム・ライフサイクル統制を可能ならしめる体制への移行には、次のような機構又は技法の確立が要請される。

- ・システムの開発及び運用を一貫して統制できる機構の確立
- ・プロジェクト管理とは異なる第三者としての開発統制技法の確保
- ・システムの保守、変更の統制技法
- ・システム寿命判定機構と基準の確保

4.4.2 開発、運用を一貫して統制できる機構

システム・ライフサイクル統制は、一つのシステムについて長期的な監視と制御を行う。したがって、プロジェクト管理とは異なり恒常的に存在し、複数のシステムや業務との対応を一元的に集約できる組織を必要とする。

図4-7は、そのような組織をもつ標準システム開発手順の例である。統制機構が『システム管理』という名称で形成されており、システム・ライフサイクル管理は『システム開発／保守／変更管理』と呼ばれている。また、その背後に、システム資源管理も恒常的な役割として組織化されている。

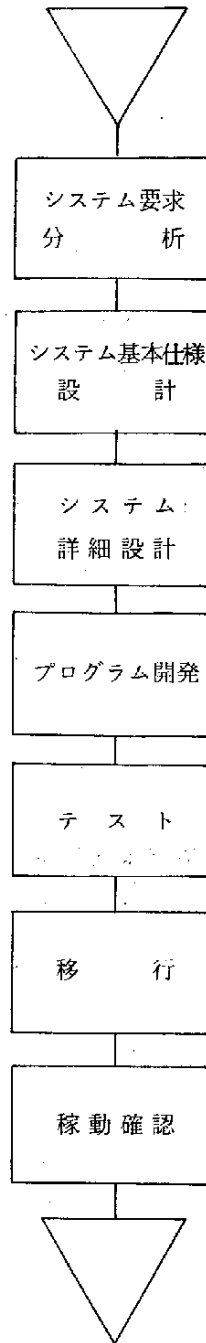


図4-5 開発過程重視のシステム・ライフサイクル

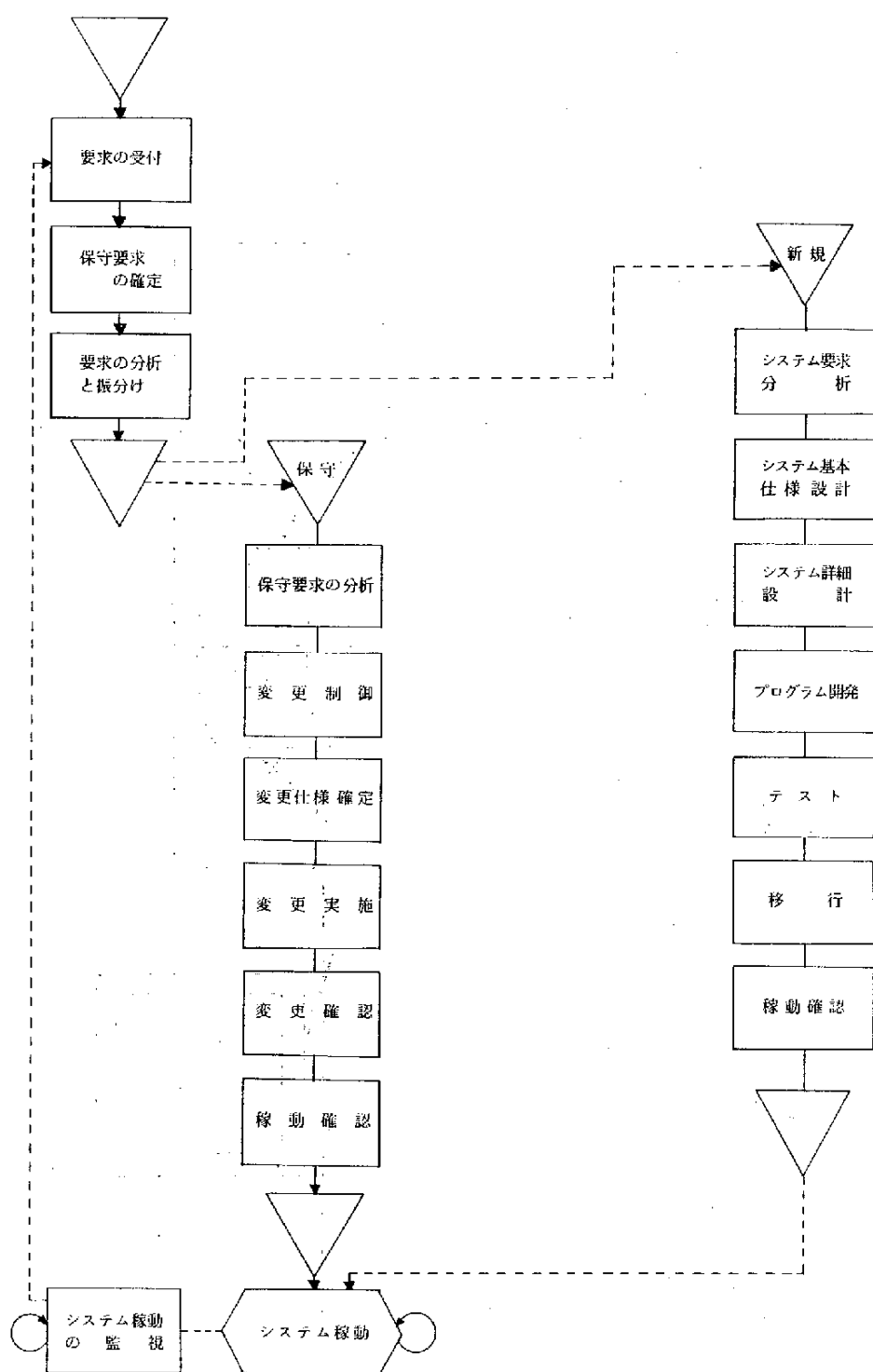


図 4-6 稼動過程重視のシステム・ライフサイクル

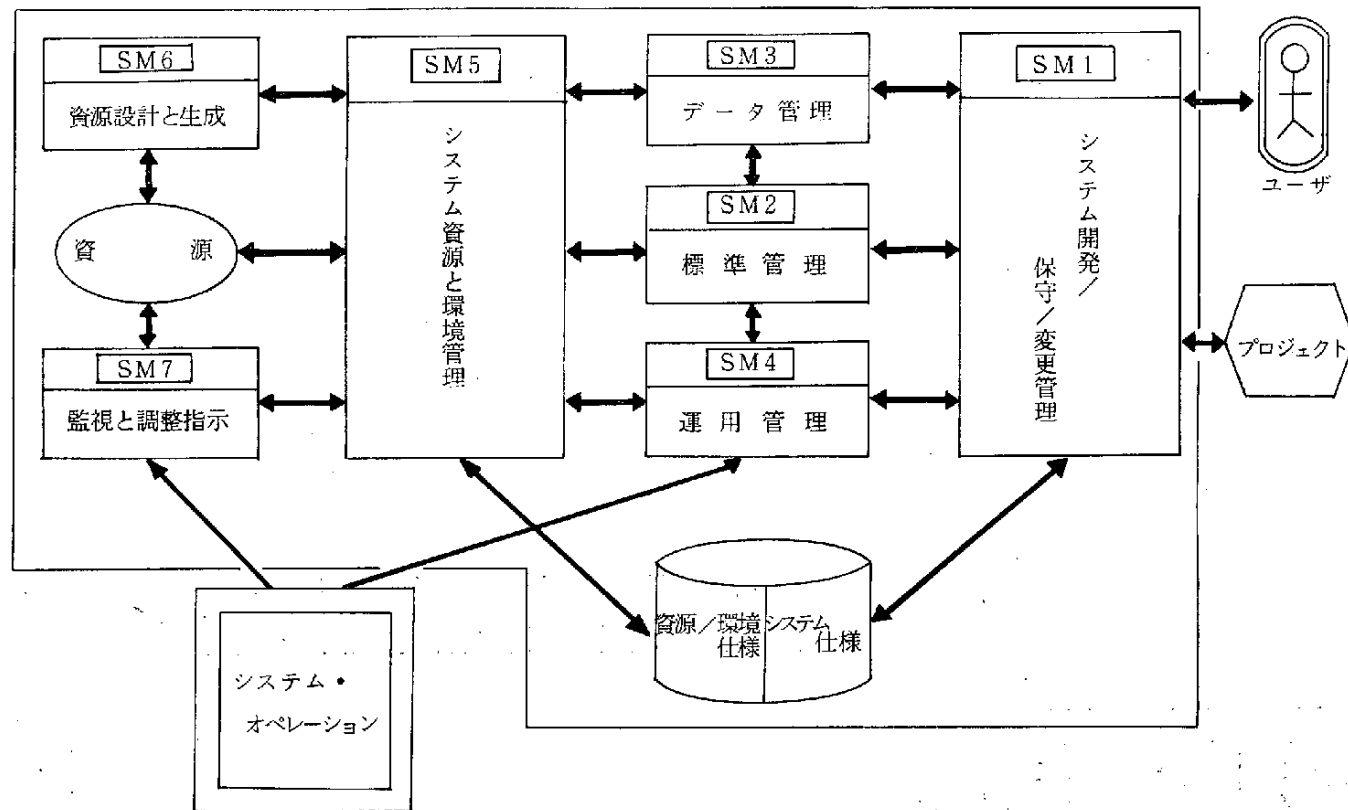


図 4 - 7 標準システム開発手順HI PACE (日立製作所)

4.4.3 開発統制としてのシステム・ライフサイクル管理

システム開発の統制については、当該システム開発に責任を持つプロジェクトの内側で実施されるものと、プロジェクトの外側にあつてシステム開発を統制するものとがある。前者は伝統的なプロジェクト管理と呼ばれるものであり、後者がシステム・ライフサイクル管理のものである。次のような役割から構成される（図4-8）。

(1) システム化要求の受付と調整

ユーザなどからの新システム開発、あるいは現行システムへの改善、変更要求を一元的に受けその重要性や必要性を吟味する。また、先に定義されているシステム全体のフレームワークや全体計画との整合性も検証する。システム一貫性を維持する上で最も重要な役割である。

(2) システム化要求の分析と事前評価

採択されたシステム化要求について、その要求をさらに分析し期待効果や開発費用の見積りなどを行う。さらに、当該要求によるシステムを実施した場合に現存する他のシステムに与える影響などを事前評価する。

(3) プロジェクトの発足準備とキックオフ

プロジェクト・イニシエーションともよばれる。プロジェクトを期待通りに進めさせるために、その目標、成果物、制約条件、体制などを定義する。これに基づきプロジェクトを発足させる。

(4) 開発過程の監査

プロジェクトの進捗中、その進め方が計画通りのものか、あるいは基準や標準に準拠したものかなどを、適切な介入点ごとに検証する。あらかじめ決めておいた介入点ごとに監査を実施する方法を、一般にSDLC（System Development Life Cycle）法と呼んでいる。

(5) プロジェクト終了処理

プロジェクト・ターミネーションとも呼ばれる。完了したプロジェクトの成果を検収し、終了手続きをとる。

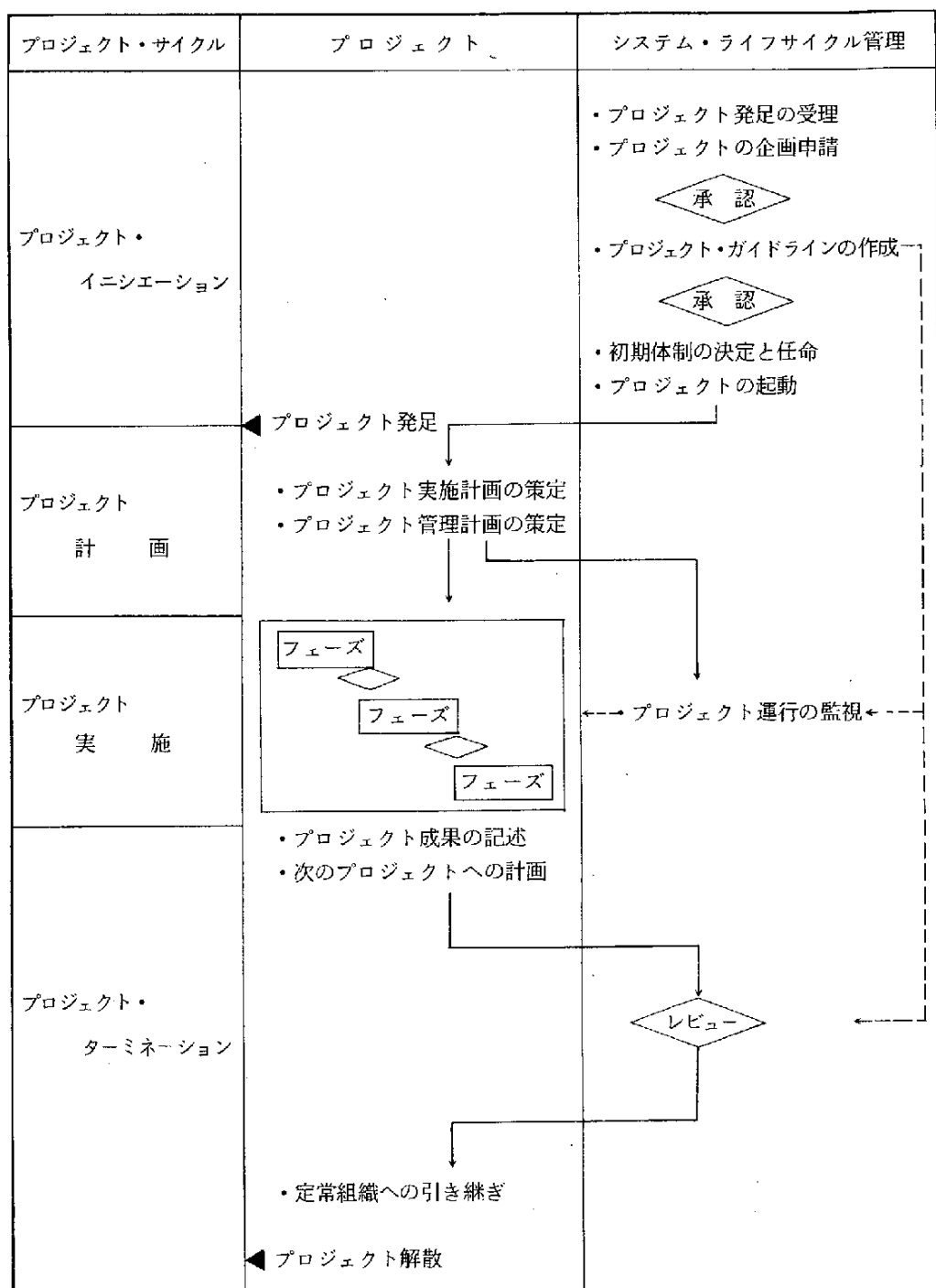


図 4 - 8 システム・ライフサイクル管理とプロジェクト管理

4.4.4 保守統制としてのシステム・ライフサイクル管理

システム・ライフサイクル管理の主たる狙いは、運用稼動状態にあるシステムの保守、変更を統制することにある。保守、変更の統制の狙いの一つは、担当者の独断によるシステム保守や変更を抑制することにある。もう一つは、システムの寿命や有効性を監視して適切に保守、変更を実施せしめることにある。

(1) 保守、変更の抑制

一つのソフトウェアは保守変更が実施される度に、その内部構造に劣化をきたす。構造の劣化は保守単価を上昇させるものとなる。上昇する保守単価は短期間のうちに新規開発単価を上回り、劣化したソフトウェアを保守するよりも新しく作り直したほうが安上がりとなる。保守費用が新規開発費用を上回る時点で、そのソフトウェアの寿命は尽きることになる。

しかしながら、現実には寿命が尽きてもそのソフトウェアの運用が継続されるだけでなく、保守費用の把握すら行われない。

一方、拡大、成長するシステムではソフトウェアの規模も飛躍的に増大する。ソフトウェア規模の増大はプログラム間の結合をも増大させ、一つのプログラム保守変更が他のプログラムの変更を呼び起こす、余波効果を大きくする。したがって、無統制な保守の実施はいたずらに保守費用の増大を招くだけでなく、システム全体の構造劣化も招来させる。

システム保守の統制は、単に恣意に任せたソフトウェア改ざんを防止するためだけでなく、システム劣化防止、保守費用削減の観点からも実施されなければならない。そのためには、言うまでもなく保守、変更の要求が直接、担当者に提示されないように、システム・ライフサイクル管理による要求の一元的受付と要求評価が不可欠となる。

(2) システム寿命と有効性の監視

システムに対する保守、変更要求は必ずしもユーザからのみ寄せられるべきものではない。ライフサイクル管理を持たないシステムでは、ユーザからの要求以外の保守変更要求は存在しない。つまり、ユーザ・クレームがただ一つの引き金である。しかし、寿命が尽きたシステムや有効性が低下したシステムを稼動、運用することは資源の無駄となるだけでなく、他のシス

テムにも悪影響を与える。

したがって、継続的にシステムの有効性を監視して、必要となる保守を適切に実施させることが重要となる。少なくとも、1つのシステムにいくらの費用を投入してきたか、また、そのシステムの運用で得られた効果の累積はいくらかを比較する仕掛け作りは不可欠である。

4.5 開発、保守の統制技法

今日、システム監査を前提としたシステム開発手順や保守手順が求められているが、個々の技法に画期的なものを求めるのは難しい。拡大期にソフトウェア・エンジニアリングとして開発された数多い在来技法の中から適切なものを選択し、それらの組み合わせの意図や手順標準化の意義に、従来になかった統制概念を盛り込むことが必要となろう。つまり、技法の運用こそ重要となる。

換言すれば、今日求められている統制は、技法の問題ではなく、企業ぐるみの組織統制の問題ともいえる。したがって、システム開発や運用に携わる管理者や経営者がどの程度、内部統制と監査の重要性を認識するかに依存しているといえる。

今後、さらに効率的な統制と監査の容易性が求められることを想定し、その実現を可能ならしめる技法を見てみたい。

4.5.1 開発統制のための技法

(1) 作業統制のための技法

ソフトウェア開発作業を統制する技法の多くはプロジェクト管理技法として開発された。プロジェクト管理技法は、その目的が不確定な環境下での課題達成におかれるところから、特に次のような工夫がなされた。これらの技法は監査の観点からも有効である。

①成果物による目標定義

漠然とした目標定義を避けるために、開発すべき物の定義を明確に行う。

②WBS (Work Breakdown Structure)

プロジェクト作業を可視的なものとするために、作業を階層構造で把握し定義する。

③フェーズド・アプローチ (Phased Approach)

大きな作業を、いくつかの単位（フェーズ：管理単位）に分割し、フェーズごとに計画（イニシエーション）とレビューを介入させる方法。フェーズはさらに小さな管理単位に分割され、同様に計画とレビューを介入させる。

このような進め方は、連続的検証（continuous validation）ともよばれ計画と実績の誤差を、できるだけ早期に検出することを狙いとしている。

監査技法としてのSDLC法はこのアプローチを前提に実施することができる（図4-9）。

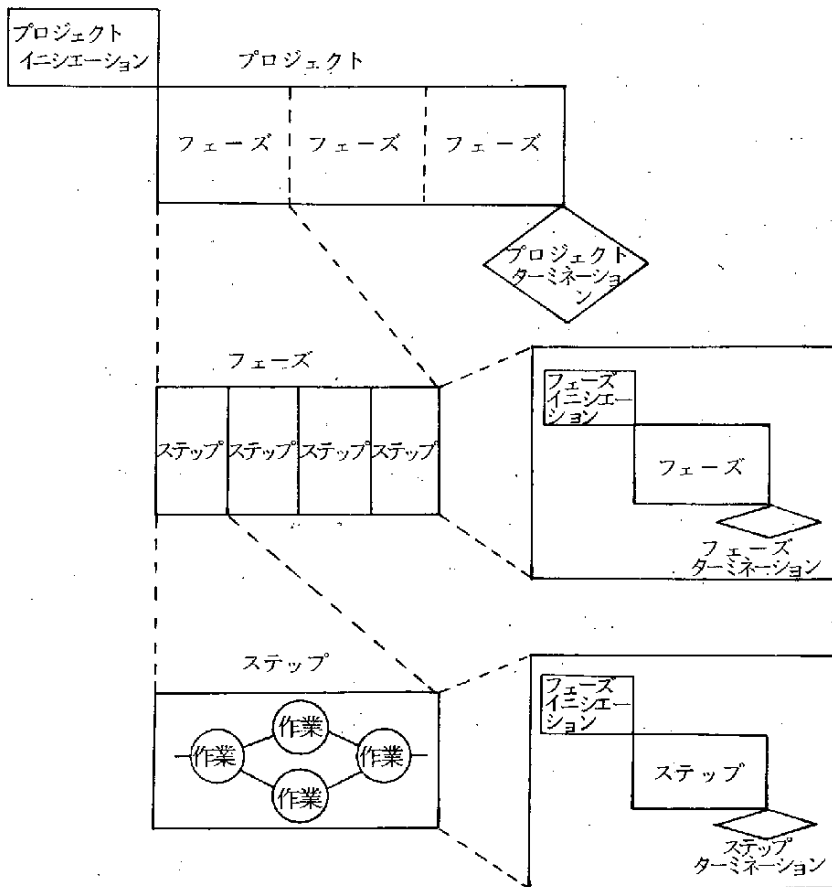


図4-9 HI PACEにおけるフェーズド・アプローチ

④作業工程の標準化

フェーズド・アプローチを前提に、作業の区分、作業内容、作業の呼び名、などの標準化を図り、混乱や不統一を最小限にするだけでなく、経験の蓄積も可能とする。

⑤作業ドキュメントの標準化

ソフトウェア・システム開発の最終成果物はプログラムである。しかし、システム開発の上流工程では、プログラムは存在しない。そこで分析、設計作業などを検証するにはドキュメントを参照するしかない。作業ドキュメントを中間成果物とみなし、WBSや標準工程と関連づけた標準化を図る。

(2) 設計、開発制約の客観化

システム開発の誤りや不整合は設計制約の曖昧性にあることは述べた。設計作業のような主観的で高度な作業を規定するために規則やルール化をこれ以上強化するのは難しい。したがって、設計作業を規定するのではなく、設計対象あるいは設計手段を規定する次のような方法が求められている。

①部品化

開発の最終製品を組み立て作業のみで実現できるような部品を先に開発、標準化して、その部品適用を統制しながら、最終製品の妥当性、正確性を保証する。これまでソフトウェアの部品化が追及されてきたが、最近では、最終製品への認識をソフトウェアではなく情報そのものとして、情報の部品であるデータの標準化を追及する動きがある。リレーショナル・データベースなどのツールが注目されている。

②ソフトウェア設計／開発支援ツール

設計や開発に伴う作業の機械化を進めながら、そのツールを使用した開発以外を認めない、あるいは設計、開発の制約をツールに埋め込むことを通じて効率的な統制の実現を図る。

③レス・プログラミング

ソフトウェア開発やソフトウェアシステムの誤り、不整合、不正などすべてソフトウェアを人間が作ることに起因する。プログラム化すべきものを最小化すればその危険性も低下する。先の部品化も同様であるが、基本的にプログラミングすべき部分を少なくすることを考える。

そのためにはシステム全体のソフトウェア構成を改善し、DBMS化、ミドルソフト化、テーブル化などを押し進めることが必要となる。

4.5.2 保守統制のための技法

システム保守の統制を有効なものとする技法、方法論には次のようなものがある。

(1) 保守抑制のための技法

無統制なシステム保守、変更を抑制するために次のような技法、または方法論が用いられる。

①保守変更要求の文書化

保守、変更に関する一切の要求を文書で記述し提出させる。これにより要求の明確化をはかるだけでなく、要求側に自発的責任意識を持たせる。

②保守変更要求の費用評価

保守、変更実施に要する費用を見積りその妥当性を評価する。特に、費用に見合うだけの効果が期待できるかの吟味が重要。一般に、新規システムの場合は費用見積りと評価を行うが、保守についてはアフターサービスの意識で受け止められ、費用見積もりも行われないことが多い。

③保守変更作業の集約的实施

一つのシステムまたはソフトウェアに対する保守要求が発生するたびに、保守作業を実施することは、当該ソフトウェアの劣化を促進させるだけでなくシステム全体の劣化を来す。一つのシステムに対する要求を幾つかまとめて実施することで劣化を緩やかなものにできる。そのためには保守要求自体を幾つかのタイプに分類することが必要となる。例えば、バグやエラー対策のように即時実施せねばならないもの、改善提案のような実施時期を繰り延べられるものに分けることである。このような対応を実現するためにもシステム・ライフサイクル管理機構が不可欠となる。

(2) 変更制御のための技法

要求された保守、変更を、その費用対効果の面から正当化できても、そ

れだけで、その妥当性を確保できたことにはならない。要求された変更が、他のソフトウェアやシステムにどのような影響をもたらすかを吟味せねばならない。場合によっては、その与える影響の大きさから要求を拒否することも必要となる。また変更を実施する場合は、与える影響範囲を確実に特定し、影響を受けるものすべてに対する確実な手配が必要となる。さらに実施された変更結果の評価、吟味も不可欠である。これらを行うための方法論、技法には次のようなものがある。

②システム構成管理 (system configuration management)

システムを構成する多数の要素 (プログラム、データ、機器など) の相互関連を管理し、その整合性を維持する資源管理方法論の一つ。管理された相互関連情報を基に、変更の影響範囲の特定を行う。

②DD/DS (Data Dictionary/Directory System)

システム構成管理を行うためのツールの一つ。特にソフトウェア・システムの構成管理に使われる。データに関する情報、データとデータ、データとプログラムなどの相互関連情報を管理する。

③再テスト手順、基準

新規開発のみならず、保守、変更されたプログラムやシステムのテストも厳密にコントロールされなければならない。新規開発のための手順や基準を定義しているところは多いが、再テスト手順を定義しているところは少ない。

④変更確認と承認

保守、変更されたプログラムが、指定された変更仕様通りのものかを確認し、承認を与える手続き。特に、変更を必要としない部分が変更されていないかの確認が重要となる。プログラムのソースコードをステップごとに比較して指定変更箇所以外が変更されていないことを確認する監査ツールも有効である。

⑤保守、変更履歴の取得

システムやプログラムごとに、何時、いかなる要求により変更を実施したかの履歴を蓄積することは、変更結果の評価、吟味に不可欠である。また、それぞれの変更時に使用したテストデータも時系列に保管することも、変更の遡及を可能たらしめる上で必要となる。

(3) システム・ライフサイクル管理のための技法

システム・ライフサイクル管理や保守統制は、システムやソフトウェアの安全性のためだけでなく、その有効性の観点からも重要である。そのためは、ソフトウェアやシステムの寿命を何等かの方法で定量化し、評価、吟味することが望まれる。しかし、そのための技法は現在、大きく立ち遅れている。次のような方法でもよいから組織内で制度化することが必要である。

①保守、変更実施回数による測定

一つのプログラムについて、その誕生以来実施された変更回数をカウントしその寿命の予想を行う。絶対的な正当化はできないが簡単な方法である。

②費用支出の累積

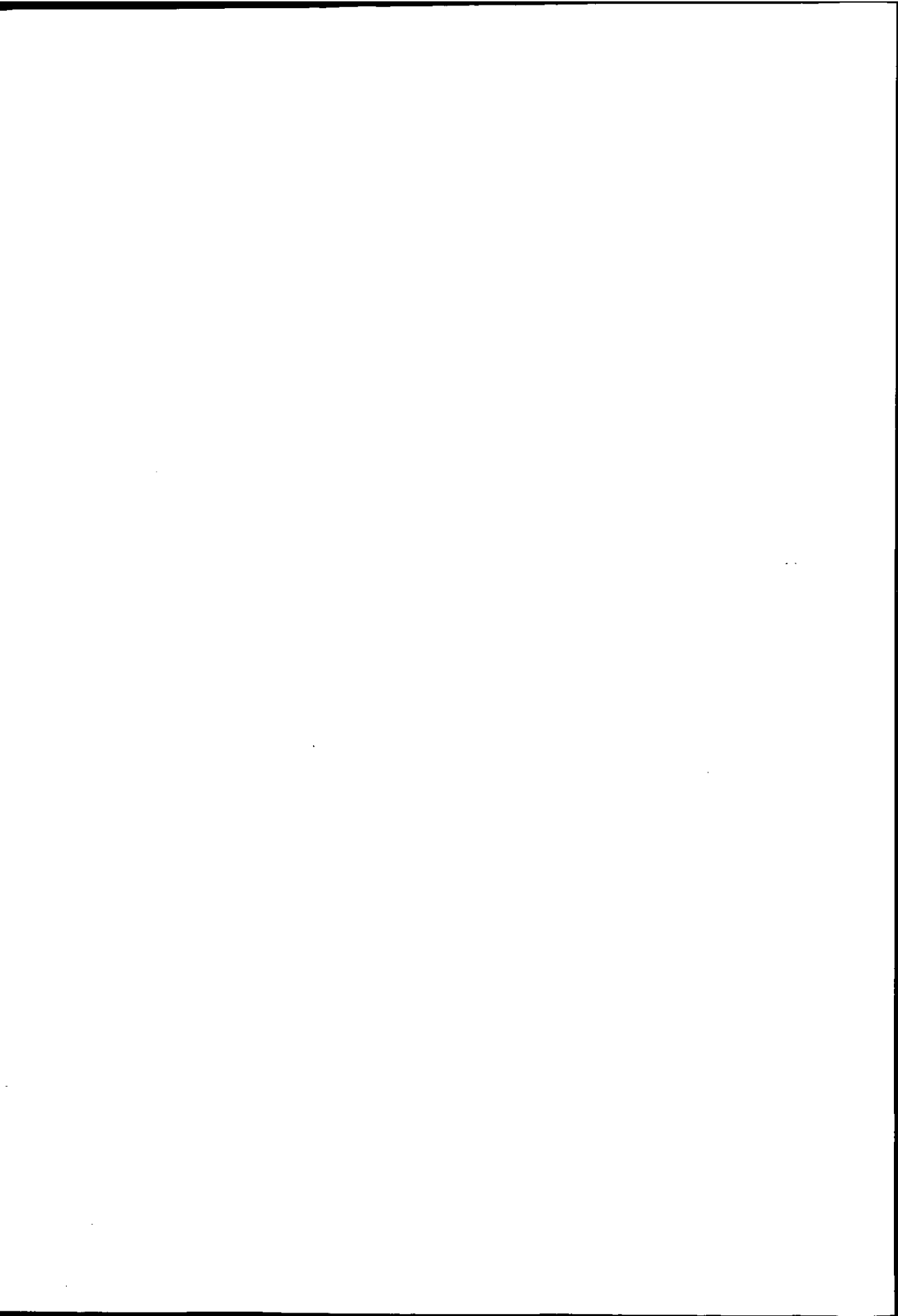
変更実施回数と同様に、誕生以来費やされた費用の累積額でその寿命を予想するもの。

③ライフサイクル・コストによる有効性尺度と評価

システムの誕生以来の累積費用と、何等かの方法で定量化した累積効果に基づき、その分岐点（回収点）を監視することが必要。数学的な厳密性よりも単純でよいから、このような管理を継続的に実施することに意義がある。

以上、情報システムのために提案された開発技法について、今日の課題である統制の確立と、そこで指導的役割を果たすシステム監査の確立を願いとして、あるべき論を述べた。

5. システム監査支援ツール・技法の試用実験



5. システム監査支援ツール・技法の試用実験

5.1 実施目的

本章で紹介するシステム監査の事例は、次の目的で実施された。

- ① 実運用しているシステムを対象にしてツール・技法を試用すること。
- ② 被監査対象に対してツール・技法の適用性を評価すること。
- ③ 適用したツール・技法の問題点を抽出すること。
- ④ ツール・技法に対する要望事項を提起すること。

なお、被監査対象の選定に当っては、A社において実運用しているシステムの中から、社内でO/A化推進の一環でシステムの改造要求が出ている“給与システム”を取りあげ、特に月例計算に目標を限定して監査することとなった。

5.2 監査対象の概要

(1) 組織の概略（A社の組織）

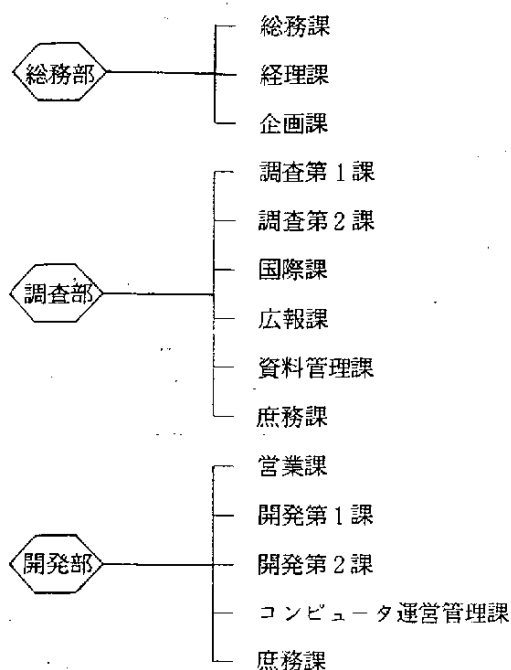


図5-1 A社の組織構成（総社員数：約150名）

(2) 対象システムの概略

<給与体系>

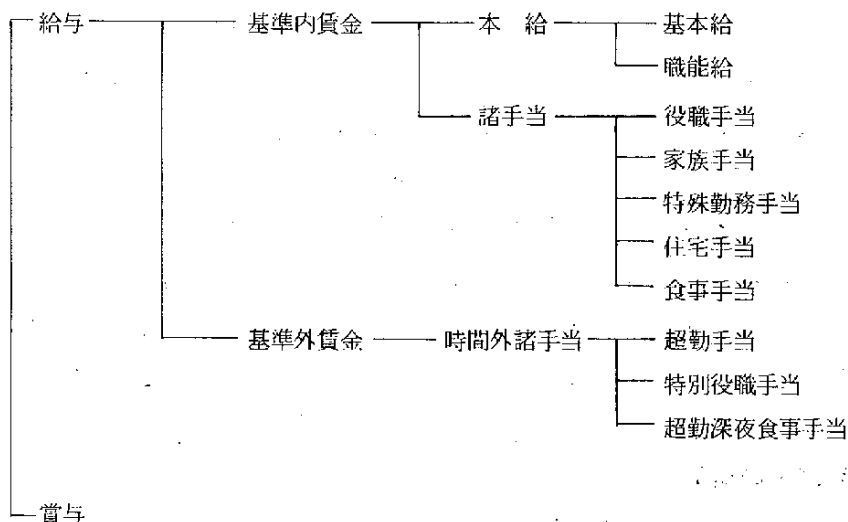


図 5 - 2 給与体系図

<給与手続き概略>

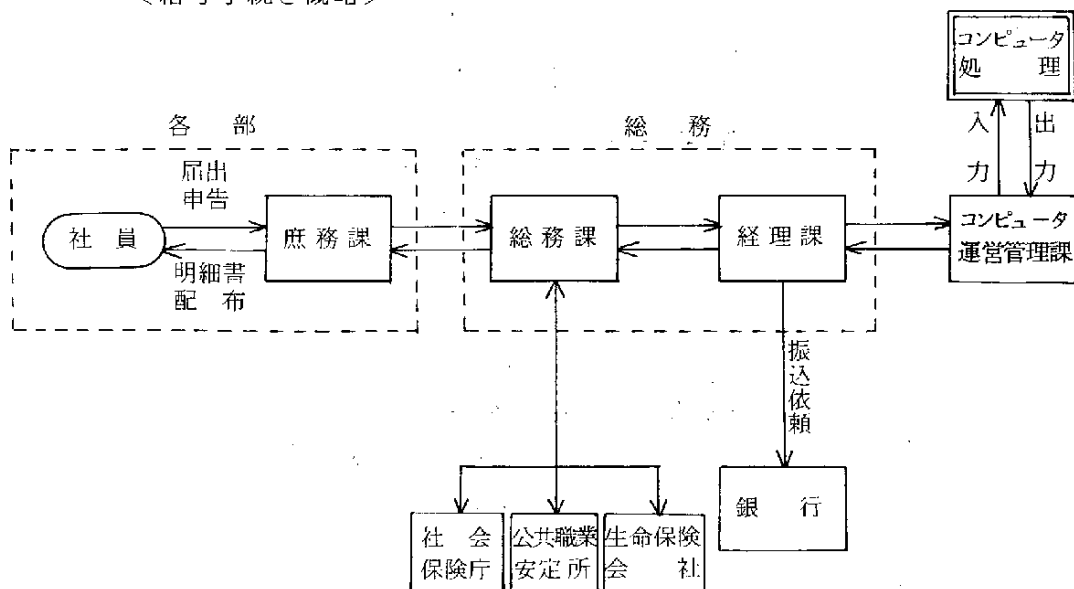


図 5 - 3 給与手続き概略図

<システム体系>

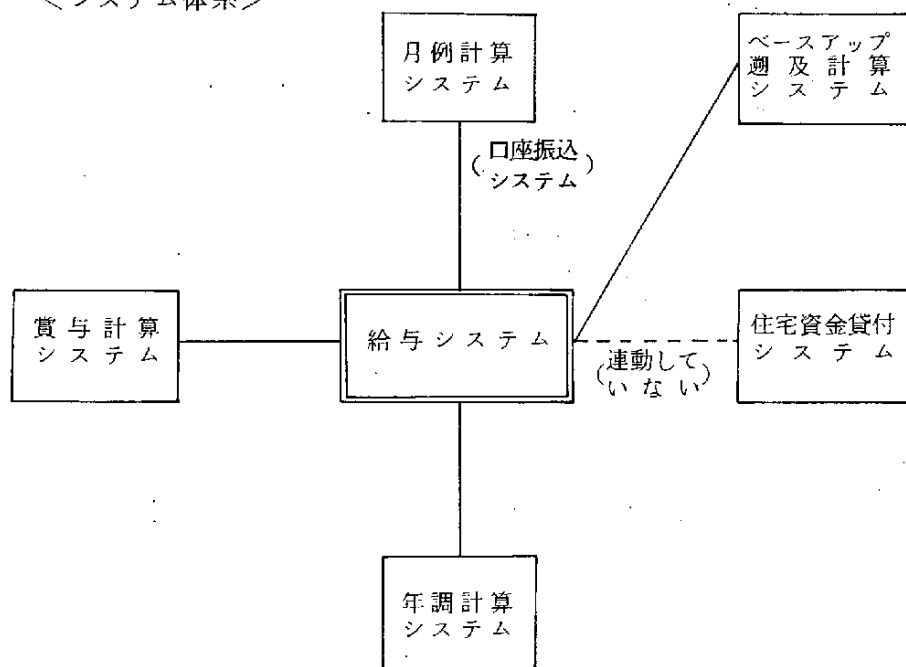


図 5-4 給与システム体系図

<ハードウェア構成>

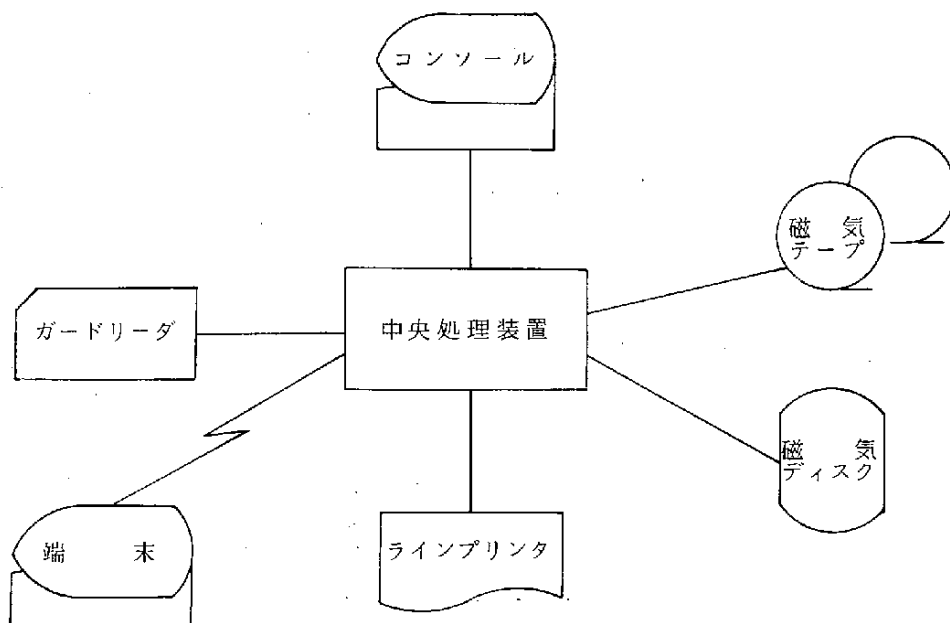


図 5-5 ハードウェア構成図

＜月例計算処理手続きフロー＞

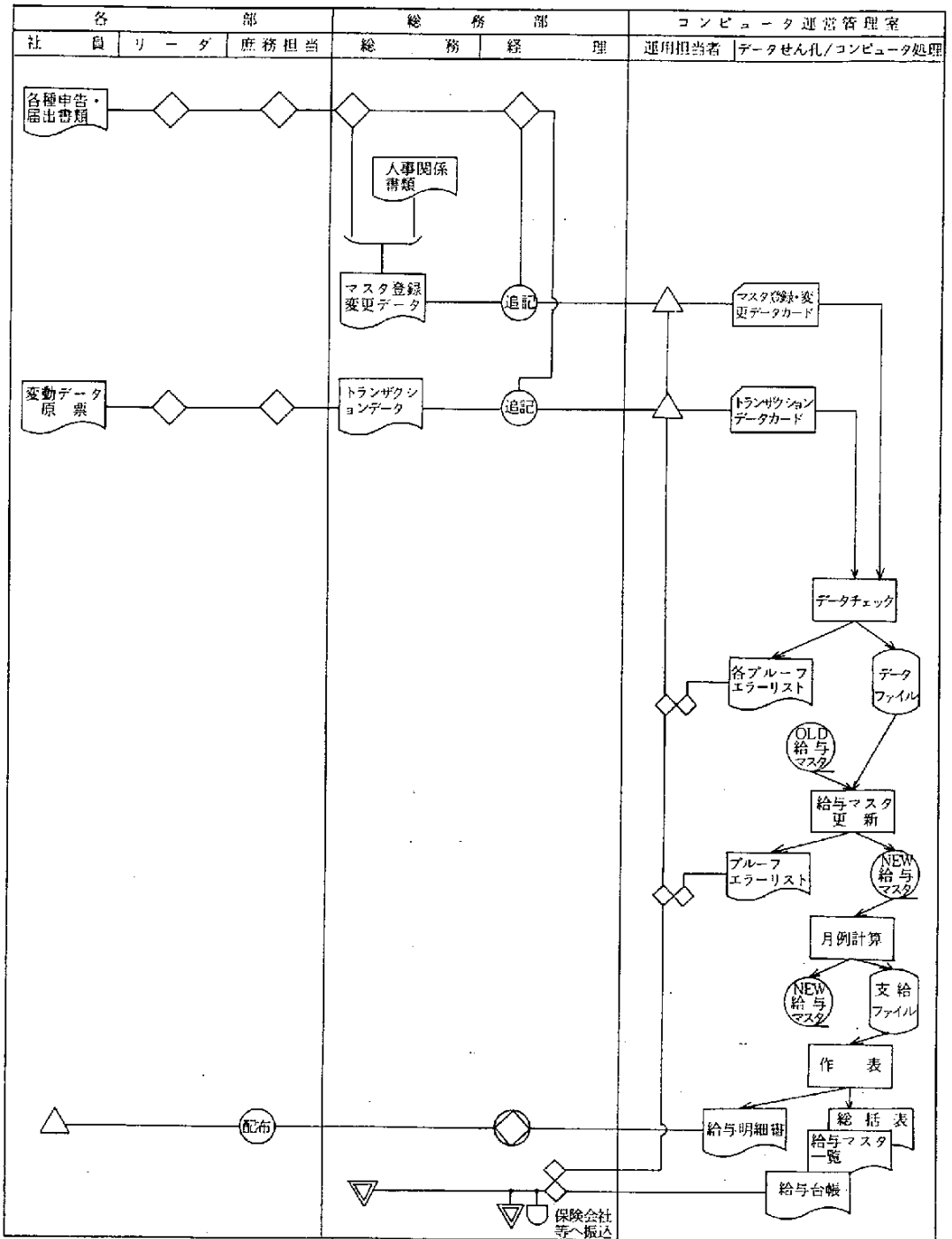


図 5 - 6 処理手続きフロー(1)

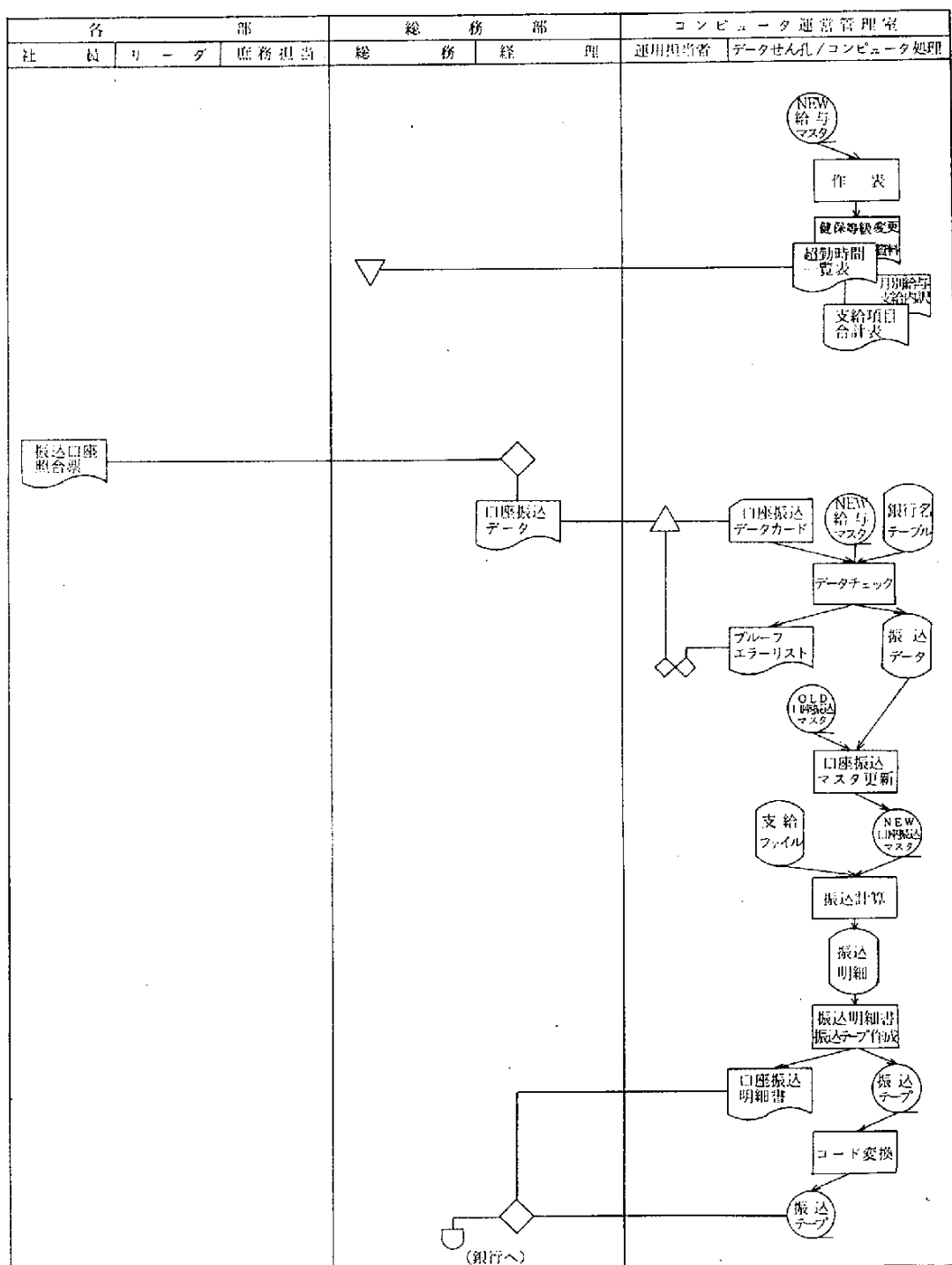


図 5 - 6 処理手続きフロー(2)

5.3 監査計画

システム監査計画の策定に当っては、「システム監査基準解説書」（日本情報処理開発協会発行）を参考にして，“システム監査基本計画書”及び“システム監査個別計画書”を作成した。

昭和 62 年 1 月 8 日

昭和 61 年度 システム監査基本計画書

1. 本年度監査の方針

当社で実運用している情報システムの中から、O A 化推進の一環で改造要求がでている給与システムを対象に監査する。

2. 監査対象と重点監査テーマ

対象システムまたは業務	重点監査テーマ
給与システム (月例計算に限定)	① 手続きの正確性 ② 手続きの信頼性 ③ 手続きの安全性 ④ 入力データチェックの正当性 ⑤ マスタ更新処理の正確性

3. 実施体制及びスケジュール

監査対象	所管部署	監査担当者	監査期間	報告時期
給与システム (月例計算)	総務課 経理課 コンピュータ運営管理課 開発第1課 各部庶務課 各部全課	(リーダ) 担当者 A 担当者 B 担当者 C	昭和62年1月 12日(月) 昭和62年2月 10日(火)	昭和62年2月 27日(金)

4. 経費予算, その他

特に支出予定無し。ただし、自社設置のコンピュータ及び周辺機器等を使用する。なお、手続きの準拠性・信頼性・安全性に関する監査手続は、所管部署担当者との面接調査、実施状況の観察、及び規程等との照合・確認作業を行うこととする。又、入力データチェックの正当性、マスタ更新処理の正確性に関する監査手続は、架空の社員(番号、氏名)による擬似給与マスタを作成した上で、各種テストケースをチェックするためのテストデータを準備し、該当するプログラムを実運用時と同様にテスト運用してプルーフリスト、ダンプリストにより確認・検証を行うこととする。

システム監査個別計画書

作成日 昭和 62 年 1 月 8 日

重点 監査テーマ	手続きの準拠性	監 対	査 象	給与システム（月例計算）	
目 的	給与に関連する諸規程及び手続きが整備され、それに基づいて正しく実施されているか監査する。	担 当 者		担当者A 担当者B	
監査項目	範 囲	着 眼 点	所管部署 担 当 者	実施日	監 査 手 続
申告・届出	・諸規程、手続きの整備 ・各種申告・届出書類への記入 ・記入内容のチェック・確認 ・書類の受け渡し ・書類の保管	・記入の正確性 ・紛失防止 ・機密保護・不正防止	・総務課 ・経理課 ・各部庶務課 ・各部全課	1/12 1/30	・諸規程・手続きが整備されているか。 ・諸規程・手続きに基づいて正確に申告・届出が行われているか。 ・記入内容のチェック・確認が適切に行われているか。 ・手続きの途中、書類が紛失しないように管理されているか。 ・書類の保管状況は十分か。
データ作成・入力手続き	・データ作成・入力手続き・要領の整備 ・データシートへの記入 ・データせん孔 ・オペレーション ・データチェック・確認	・正確性 ・不正防止	・総務課 ・経理課 ・コンピュータ運営管理課	1/12 1/30	・データ作成・入力手続きが整備されているか。 ・データ作成・入力の各種要領が整備されているか。 ・データ作成・入力が行われているか。
プログラム作成・変更	・プログラム作成・変更手続きの整備 ・プログラム作成・変更後のチェック・確認 ・作成・変更に伴うドキュメントの整備	・正確性 ・不正防止	・総務課 ・経理課 ・コンピュータ運営管理課 ・開発第1課	1/12 1/30	・諸規程・制度の変更に伴うプログラム作成・変更手続きが整備されているか。 ・プログラム作成・変更後のチェック・確認が十分行われているか。 ・プログラム作成・変更後のドキュメントは整備されているか。

システム監査個別計算書

作成日 昭和 62 年 1 月 8 日

重点 監査テーマ	手続きの信頼性		監 対	査 象	給与システム（月例計算）	
目 的	各人が申告・届出したデータ、及び総務・経理課が生起したデータが正しく作成されているか監査する。		担 当 者		担当者 A 担当者 B	
監査項目	範 囲	着 眼 点	所管部署 担 当 者		実施日	監 査 手 続
データ作成・ 入力	・データシートへの記入 ・記入内容のチェック・確認 ・データせん孔 ・データチェック・照合	・記入内容の正確性・網羅性 ・入力データの正確性	・総務課 ・経理課 ・コンピュータ運営管理課		1/12 ? 1/30	・原票（申告・届出書類等）からデータシートへ正しく記入されているか。 ・データシートから正しくデータせん孔され、データチェック照合作業が適切に実施されているか。

システム監査個別計算書

作成日 昭和 62 年 1 月 8 日

重 点 監査テーマ	手続きの安全性		監 査 対 象	給与システム（月例計算）	
目 的	給与に関する個人のプライバシー（機密）が侵害されないように保護されているか、不正防止対策が図られているか等、管理・統制面を監査する。		担 当 者	担当者 A 担当者 B	
監査項目	範 囲	着 眼 点	所管部署 担 当 者	実施日	監 査 手 続
マスタ登録・ 変更手続き	- データシートへの記入 - データせん孔 - オペレーション - 原票・データシート・データカードの受け渡し	- 機密保護 - 不正防止	- 総務課 - 経理課 - コンピュータ運営管理課	1/12 ？ 1/30	- 人事関係書類等からデータシートへの記入作業中、及びデータせん孔中、オペレーション中に個人のプライバシーが漏れないように管理されているか。 - 同作業中に改ざん等の不正を防止できるように管理されているか。
書類及び出力 帳票類の保管	- 書類の保管 - 出力帳票類の保管 - 不要書類の廃棄 - 不要帳票類の廃棄	機密保護	- 総務課 - 経理課 - コンピュータ運営管理課	1/12 ？ 1/30	- 人事関係書類の保管方法が定められ、それに基づいて管理されているか。 - 出力帳票類の保管方法が定められ、それに基づいて管理されているか。
カード／ファ イル管理	- データカードの管理 - JCLカードの管理 - ディスクの管理 - 磁気テープの管理 - パスワードの管理	- 機密保護 - 不正防止	コンピュータ運営管理課	1/16 ？ 2/10	- データカード、JCLカード、ディスク、磁気テープの管理方法が定められ、それに基づいて管理されているか。 - 機密保護、不正防止対策が図られているか。

システム監査個別計画書

作成日 昭和 62 年 1 月 8 日

重点 監査テーマ	入力データチェックの正当性		監 対	査 象	給与システム（月例計算）	
目 的	データ処理の中で、入力データの 妥当性チェックが正しく実施され ているかを監査する。		担 当 者		担当者A 担当者C	
監査項目	範 囲	着 眼 点	所管部署 担 当 者	実施日	監 査 手 続	
入力データチ ェック機能	・フォーマット チェック 〔データチェ ック・プロ グラム〕 ・給与マスタと の照合チェッ ク 〔給与マスタ 更新プログ ラム〕	・チェックの 妥当性 ・チェックの 正確性	・開発第1課 (開発担当)	2/2 1 2/10	・正データ、誤デー タをテストデータとし て入力した場合、チ ェックが正しく行わ れるか。 〔入力データ、ブル ーフ/エラーリスト、 給与マスタファイ ルの内容を比較〕	

システム監査個別計算書

作成日 昭和 62 年 1 月 8 日

重点 監査テーマ	マスタ更新処理の正確性		監 対	査 象	給与システム（月例計算）	
目 的	データ処理の中で、給与マスタ変更指示に従って給与マスタの内容が正しく更新されるかを監査する。		担 当 者		担当者A 担当者C	
監査項目	範 囲	着 眼 点	所管部署 担 当 者	実施日	監 査 手 続	
マスタ登録・ 変更処理機能	- 給与マスタ更新処理 〔給与マスタ更新プログラム〕 - 月例計算処理 〔月例計算プログラム〕	処理の正確性	開発第1課 （開発担当）	2/ 2 と 2/10	入力データチェックを経たテストデータを用いて、給与マスタ更新処理及び月例計算処理を実行した場合、給与マスタファイルが正しく更新されるかチェックする。 〔更新前後の給与マスタの内容を比較〕	

5.4 監査の実施

(1) 実施方法

本監査を実施するに当たっては、“システム監査基本計画書”及び“システム監査個別計画書”に基づいて、表5-1に示すような方法で行った。以下に概要を述べる。

なお、本調査を開始する前に、予備調査を行って情報を収集するとともに各所管部署担当者に対するインタビュー（質問）用にチェックリスト／質問書を作成する作業を行った。

表 5 - 1 監査の実施方法

重点監査テーマ	実 施 方 法 （主なもの）
① 手続きの準拠性	・視察（現場の状況を観察） ・質問（チェックリスト／質問書を用いて担当者より実状を調査）
② 手続きの信頼性	・実査（実際の作業内容を検査） ・視察（現場の状況を観察）
③ 手続きの安全性	・実査（実際の保管状況等を検査） ・立合い（手続きの流れに従って視察） ・質問（チェックリスト／質問書を用いて担当者より実状を調査）
④ 入力データチェックの正当性	・当監査目的プログラム（マスタ内容表示プログラム） ・ユーティリティ（ファイル・ダンプ） ・テストデータ法（マスタ登録・変更データ、トランザクション・データの正・誤データ）
⑤ マスタ更新処理の正確性	・当監査目的プログラム（マスタ内容表示プログラム、フォーマット変更プログラム） ・ユーティリティ（ファイル・ダンプ、ファイル・コンペア） ・テストデータ法（マスタ登録・変更データ、トランザクション・データの正データ）
	（備考）・当監査目的プログラムとして、他に2本（擬似給与マスタ作成プログラム、データファイル初期化プログラム）を作成し④、⑤の実施前に利用した。

① 手続きの準拠性

準拠性の監査では、根拠となる諸規程や手続きの整備状況を調査した上で、それらに基づいて正しく実施されているかを監査する方法をとった。

諸規程及び手続きの整備状況については、概ね予備調査で情報収集を行っていることから、本調査ではチェックリスト／質問書に従って再確認と補足事項を聴取する方法により行った。

準拠性監査そのものについては、実地現場（データ記入場所、データせん孔場所等）の状況を観察し、諸規程や手続きに定められた通り実施しているか検査する方法をとった。

② 手続きの信頼性

信頼性の監査では、各人が申告・届出したデータ、及び総務や経理で生じたデータが正しく作成されるかを監査した。そのため、原票（申告・届出書類等）に記述されている内容の通りに正しくデータシート（マスタ登録・変更データシート、トランザクション・データシート）へ記入されているかを実査する方法で行った。

実査では、データシートへの記入担当者が作業（集計・転記）を終えて責任者（上司）のチェックが終了した後、監査担当者が再度同作業を行って確認する方法をとった。

データシートへの記入作業以降のチェック・確認、データせん孔／ベリファイ、入力データのプルーフ／エラーリストのチェック・照合作業については、作業状況を各実地現場で視察する方法で行った。

③ 手続きの安全性

安全性の監査では、個人の機密情報が漏洩しないように保護対策が図られているか、及びデータ改ざん等の不正防止対策が図られているかを監査した。そのため、機密書類やカード／ファイルの管理方法が定められているか調査した上で、それらに基づいて、あるいは他の方法で運用されているか実査する方法により行った。

機密保護対策、不正防止対策の管理方法の規程については、手続きの準拠性監査と同様に整備状況を確認する方法をとった。

実際の運用状況については、手続きの流れに従って立合い／視察を行う

とともに、現場の管理状況を実査することで行った。

④ 入力データチェックの正当性

入力データチェックの正当性監査では、入力データの妥当性チェックが正しく行われているかを監査することにある。そのため、各種のケースを想定したデータ（マスタ登録・変更データ、トランザクション・データそれぞれにつき正データと誤データ）を用意し、データチェック・プログラム及びマスタ更新プログラムのチェック機能を検査した。

当監査のため、ユーティリティ（ファイル・ダンプ機能）を利用するとともに、マスタ内容表示プログラムを作成して利用した。

実際の検査では、テストデータとプルーフ／エラーリスト、及び給与マスタのダンプリスト等を比較・照合することにより確認を行った。

⑤ マスタ更新処理の正確性

マスタ更新処理の正確性監査では、マスタ登録・変更データの指示に従ってマスタの内容が正しく更新されるかを監査することにある。そのため、各種のケースを想定したマスタ登録・変更データの正データを用意し、マスタ更新プログラム及び月例計算プログラムによるマスタ更新機能を検査した。

当監査のため、ユーティリティ（ファイル・ダンプ、ファイル・コンベアの両機能）と、マスタ内容表示プログラム、フォーマット変更プログラムを作成して利用した。

実際の検査では、テストデータと、給与マスタの更新前後のダンプリスト等を比較・照合することにより確認を行った。

(2) 実施手順

重点監査テーマ④と⑤について、実施手順をフローにして図5-9、図5-10に示す。なお、④、⑤実施前に図5-7、図5-8に示す処理を行っておく。（フロー中、プログラム枠右下角に▲印がついているのが当監査用に特別に作成したプログラムである。以降も同様。）

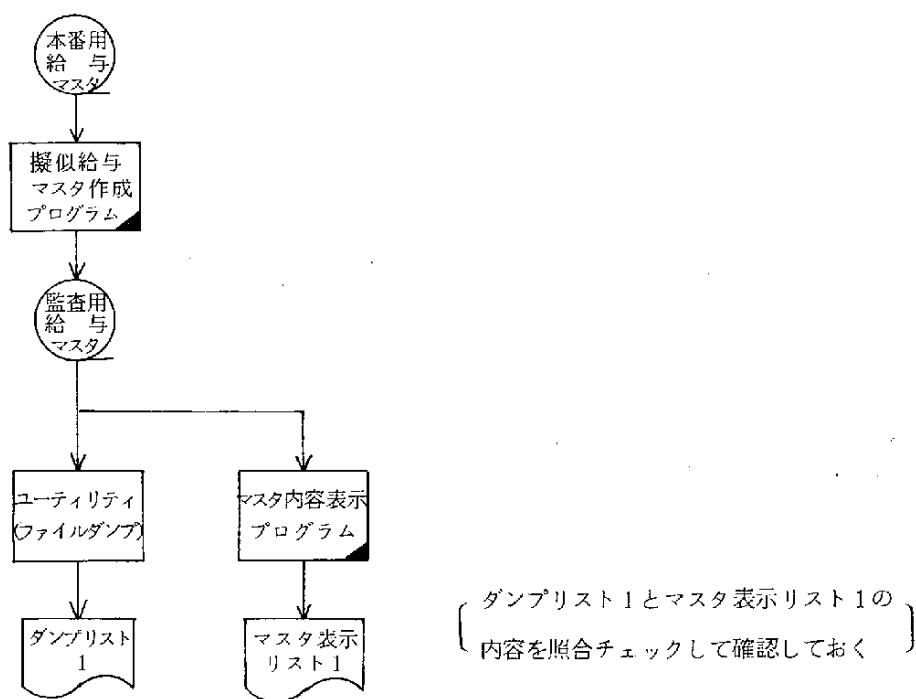


図 5 - 7 実施手順 - 1 (監査用給与マスタ作成)

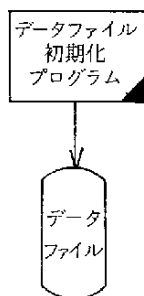


図 5 - 8 実施手順 - 2 (監査用データファイル初期化)

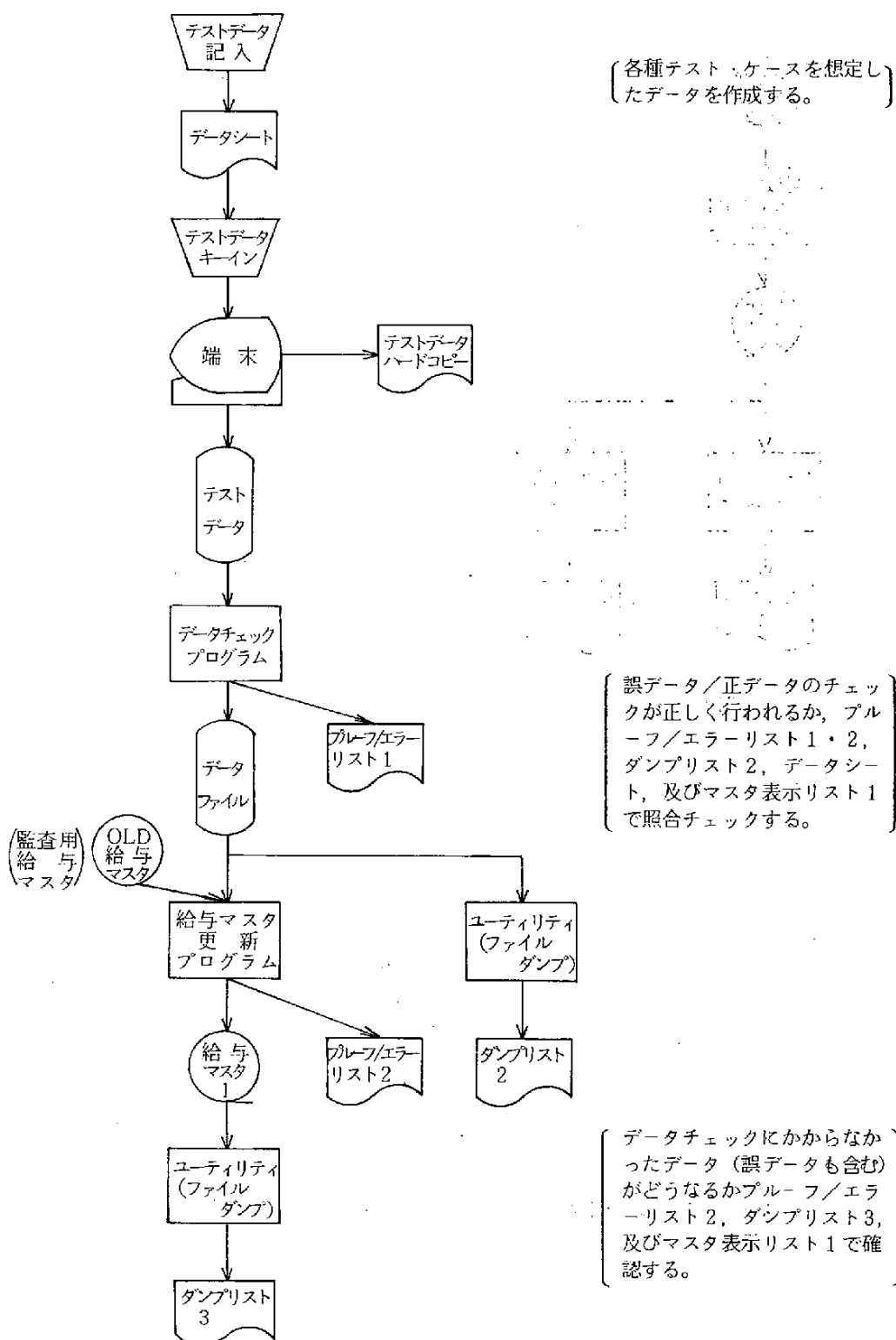


図5-9 実施手順-3 (入力データチェック正当性監査のための手順)

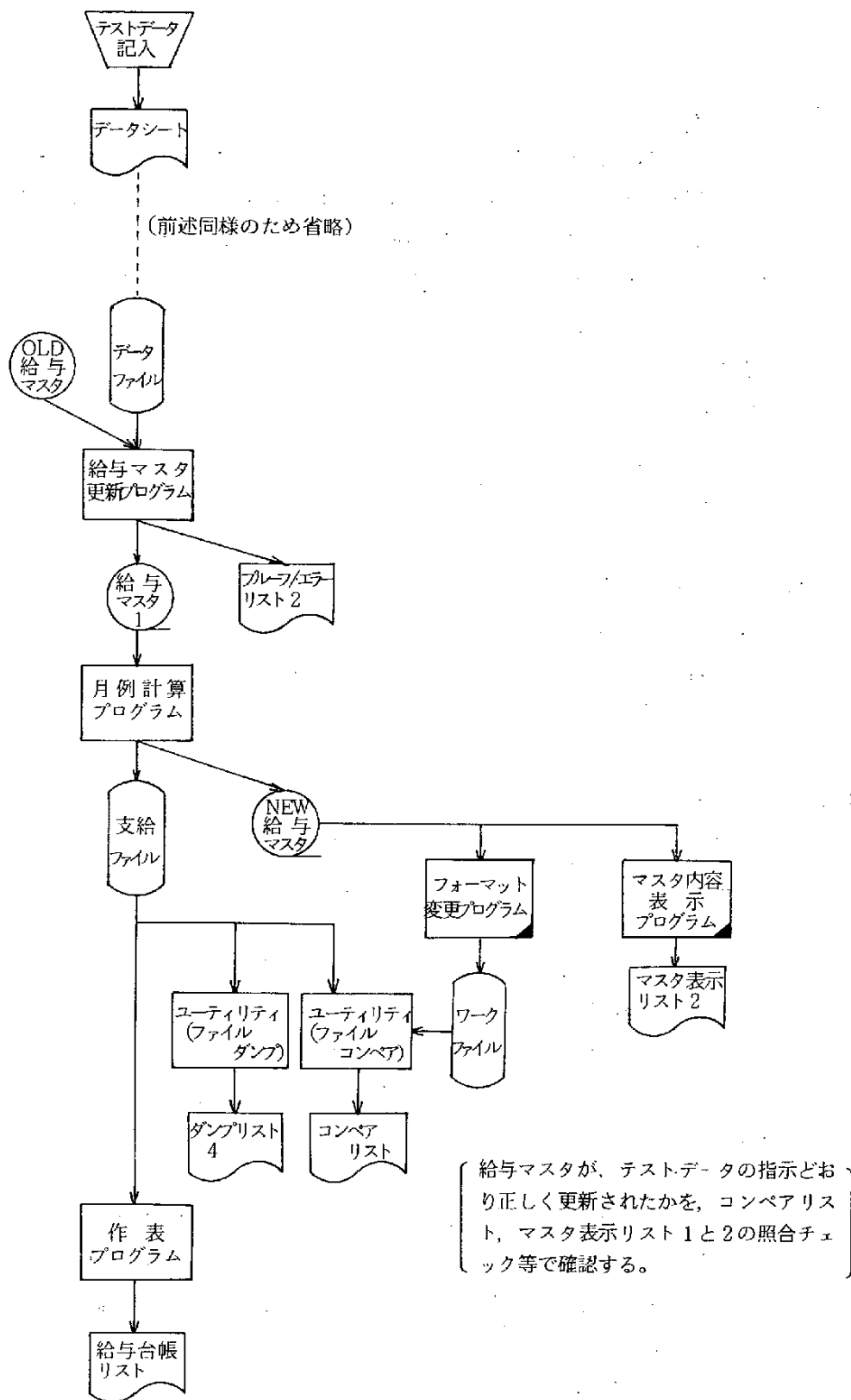


図 5 - 10 実施手順 - 4 (マスタ更新処理の正確性監査のための手順)

5.5 監査報告

システム監査報告書の作成に当っては、前出の「システム監査基準解説書」を参考にして様式を決定し、監査結果のとりまとめを行った。又、本調査結果の全容（詳細）については“システム監査調書”として別途とりまとめた。

なお、システム監査報告書には以下に示す項目についてそれぞれのテーマごとにとりまとめを行った。

- (1) 重点監査テーマ
- (2) 目 的
- (3) 範囲・手続きの概要
- (4) 所管部署
- (5) 概要（総合評価）
- (6) 指摘事項（問題点の指摘）
- (7) 改善勧告
 - ① 緊急改善
 - ② 通常改善

5.6 評価考察

- (1) 被監査対象に対するツール・技法の適用性

被監査対象（給与システム）に対してどのようなツール・技法を使用あるいは検討したのか以下に簡単に述べる。

① 手続き面の監査技法

重点監査テーマのうち“手続きの準拠性、信頼性、安全性”に関する監査では、いわゆる“伝統的監査技法”と呼ばれる手法を検討し、いくつかを採用した。

イ. 実 査

監査担当者がカードや磁気テープあるいは書類・帳票類の保管場所に直接行って所在の確認をする際に用いた。また、集計・転記作業の結果が正しいかどうかを、監査担当者が実際に同作業を行って確認することもこの手法に類する。

ロ. 視 察（観察）

監査担当者が作業現場に赴いて、作業状況（データ記入作業等）を観

察することがこの手法に属する。チェックリストに基づいた観察も行った。

ハ. 立合い

監査対象の手続きに従って実施されている状況を監査担当者が観察し、規程（あるいはそれに類する慣例部分を含む）通りに行われているか適否を確認する際に用いた手法である。

ニ. 突合せ

原票（申告・届出書類）とデータシート、あるいは原票とブルーフ／エラーリストとを照合・確認するのに用いた手法である。

ホ. 質問／インタビュー

監査担当者が被監査部門の業務担当者と面接し、実状を調査した手法である。チェックリストや質問書を作成しておき、それに基づいて質問を進める方法もとった。

ヘ. 監査手引き法

今回の監査では、チェックリストやガイドラインの標準が存在しなかった。又、監査手順も無い状況でスタートしたため、“システム監査基準解説書”や他の文献を参考にして、当監査用に簡単なチェックリスト相当のものを作成して監査を実施した。

ト. 統制流れ図法

予備調査を実施した段階で、給与システムの手続きフローが存在しないことが判明したため、当手法は適用不可能となった。

② 処理面の監査支援用ツール・技法

重点監査テーマのうち“入力データチェックの正当性”と“マスタ更新処理の正確性”に関する監査では、コンピュータを利用した監査支援用ツール・技法のいくつかを採用したが、検討したものも含めて以下に述べる。

イ. テストデータ法

監査対象としたプログラム（データチェック・プログラム、給与マスタ更新プログラム、月例計算プログラム）の処理の正確性を検証するための技法である。各種テストケースを想定したテストデータを作成し、本番のプログラムを実行して処理結果の正当性を評価するのに利用した。

ロ. 総合テスト法（I T F 法）

この技法によればコンピュータ処理前後の手続きを含めた正確性や完全性の検証を得ることができる。しかしながら当監査で対象としたプログラムがシステムのほんの一部であることと、監査対象としてデータチェック機能部分に重点をおいたため、前述のテストデータ法で十分であると判断し採用しなかった。

ハ. 平行シミュレーション法

シミュレーションの実施は、そのシミュレーション・プログラムの作成工数、あるいは給与計算パッケージの購入等の負担に比して、いずれも有効性が十分ではないと判断したため採用しなかった。

ニ. 汎用監査ソフトウェア法

汎用監査ソフトウェアは一般的に高価（数百万円以上のものが多い）であり、監査対象の規模に比して購入するのは不相当と考え採用しなかった。

ホ. 特殊目的監査プログラム法

今回の監査対象システム用に、監査人が特別に作成したプログラムを利用した。主にユーティリティの機能不足をカバーするためのものである。

ヘ. ユーティリティ

メーカ提供のユーティリティから、ファイル・ダンプ、ファイル・コンペアの機能を利用した。

(2) 適用したツール・技法の問題点

当監査で実際に適用したツール・技法について指摘事項を以下に述べる。

① 手続き面の監査技法

“実査”、“立合い”については監査人自らの作業となるので、被監査部門の業務担当者に対して何の負荷もかからないことからスムーズに実施することができた。

“視察”については、業務担当者の作業状況を近くで観察するため、通常の作業遂行時とどうしても異り、監査されているという意識が強くなる。その影響のためか、いつもより念入りに作業している様子もうかがえるし、又、いつも間違いがなかった所で記入ミスが生じたりもしたようだ。

“質問／インタビュー”は、業務担当者に時間を割いてもらって実施するためその調整が必要であった。

その他、チェックリストが十分なものでなかったことから、要領よく質問／インタビューができず、後日電話で再確認をする等不備な面も明らかになった。

② 処理面の監査支援用ツール・技法

イ. テストデータ法

テストケースを設定するには、給与システムの機能を十分理解しなければ不可能である。そのためテストデータ作成作業以前に、予備調査に多大な時間を要することになった。

チェックリストの作成に当たっても、データ記入要領を基にしてチェック項目（監査項目）を網羅的に挙げていく方法をとらざるを得なかった。

次に、チェックリストに従ってテストデータを作成する段階では、正データの作成はデータ記入要領に適合したデータを作成すればよいので容易であるが、誤データの作成をどの範囲まで行うべきか判断しにくい所があった。たとえば範囲チェックのための誤データを作る場合、範囲の両端の値をとればよいのか、範囲からかけ離れた値をとる必要があるのか迷う所である。妥当性のチェック用のデータでも、どのくらいの値が妥当といえるのか不明瞭な項目もある。今回の監査対象が給与システムという特殊性もあったであろうが、チェック項目が多く、有限なテストデータ量で行ったので、必要十分なデータがほんとうに網羅されたのか確信が持てない。

ロ. ユーティリティ

ユーティリティは開発用のツールの一つであることから、監査人がチェック・確認のために用いるには不満は当然のことといえる。出力されるリストは見にくく、機能が貧弱であるうえに、使用方法を理解するにも、監査人がマニュアルを解説するのは容易でない。

ハ. 特殊目的監査プログラム法

ユーティリティの機能不足をカバーするために、当監査対象用の監査プログラムを作らざるを得なかったが、プログラム作成に意外と時間を要してしまった。単一目的の監査プログラムの作成には限界ある

と痛感した。

(3) ツール・技法に対する考察

当監査で適用してみたシステム監査支援用のツール・技法については前述した通りである。ここでは適用してみたツール・技法も含めて、ツール・技法に“こんな機能があったら”と思われたことを以下に述べることにする。

- ① 監査経験の浅い人が、監査を実施しなければならなくなった時、現状では“システム監査基準”や“システム監査ガイドライン”等を参考にして、実施計画の策定から監査の実施、監査報告書の作成まで、いわゆる監査手続きを確立することから始めなければならない。このような時に要求されるのが「監査手続き」を支援してくれるシステムである。システム監査の専門家（監査経験が十分な人）の知識を基にして知識ベース化したり、業務毎の専門知識を知識ベース化して、エキスパート・システムから支援できるようになれば効果的であろう。

監査手続きの個々については、次のような機能が支援されると有効と思われる。

イ．システム監査計画を支援する機能

ロ．システム監査計画に基づいて監査の実施状況を管理・統制するための機能

ハ．質問書／チェックリストを業務毎に支援する機能

ニ．監査情報を分析し、監査調書、システム監査報告書の作成を支援する機能

- ② 処理の正確性を検証するための一つの技法であるテストデータ法は、一般的にテストデータの作成に膨大な労力と時間を要する欠点がある。又、有効なテストデータを作成するには、対象とする業務システムについて詳細な知識を有することが必要なことも欠点である。その解決策の一つとして、テストデータの作成を支援するツール、いわゆるテストデータ・ジェネレータが要求される。（できるだけ業務システムの知識を必要としないようにする。）正データ、誤データ、項目間の関連性をチェックできるデータ等複雑なデータまでを作成し、マッピング法と連動して半自

動的にテストできるような機能が望まれる。

- ③ メーカーから提供されるユーティリティは、システム監査と目的を異にして用意されているツールのために、使い勝手は悪く、機能も不十分であり、しかも利用するためのマニュアル類は概して使いにくいと不評である。これらユーティリティ機能の充実と操作性の向上を図ってシステム監査用にしたのが汎用監査ソフトウェアとも言えよう。ただし商品化されている汎用監査ソフトウェアといえども十分な機能と操作性のよいものは少ないと言われており、今後さらに改良された（機能の充実と操作性の向上を図った）汎用監査ソフトウェアやユーティリティ等が望まれる。
- ④ 最近、一部の汎用監査ソフトウェアに、日本語の処理機能を加えたものや、端末からオンラインで操作できるようになったものが出現してきたが、今後さらにこのような機能等を充実させることが望まれる。

6. システム監査支援ツールの課題

6. システム監査支援ツールの課題

6.1 現在のシステム監査支援ツールの課題

これ迄にも数多くのシステム監査支援ツールが開発され、実用に供されてきたが、必ずしもオールマイティのものではなく、いずれも一長一短の特色を有しているのが実情である。

以下に、これ迄のシステム監査支援ツールに対する問題点と今後の課題について見てみよう。

6.1.1 機能上の課題

(1) データベース処理機能のレベルアップ

第一は、既往のシステム監査支援ツールが、主としてバッチ・ベースの処理を主体としたものが多かっただけに、一般にデータベースへのダイレクトなランダム・アクセスがしにくいものが多いことである。そのために、オンライン・リアルタイムで異常値の発見を行いたいような場合にも、いったんシーケンシャル・ファイルにおとしてからチェックを行っているケースが多い。

最近では、システムが大規模化すると同時に、処理内容も高度化・複雑化し、データベースを前提としたシステムが増えているだけに、監査目的に必要な機能をDD/DSに追加することによって、システム監査自体の有効性や効率性の向上をはかることが望まれているといってもよさそう。

特に、オンライン・リアルタイム・システムについては、組込み監査モジュール法なども有効な手法ではあるが、ITF法などについては、その重要性が今後いっそう高まっていくと思われるにもかかわらず、汎用の監査ソフトウェアの開発がなかなか難しい側面を持っている。そのようなこともあって、リモートのパソコン端末からパラメータをインプットして、遠隔地のメインフレームのデータベースや汎用監査ソフトウェアを動かしたいという要望も現実に出されている。

(2) ユーザ・フレンドリーなシステムの開発

内部監査人は、現状ではまだ一部の例外を除き、コンピュータ・システムにはなじみの少ない人が多い。それだけに、既存の汎用監査ソフトウェアでは、JCLのパラメータのセットなどをわずらわしく感ずる向きが多

く、現実にもかなりの JCL のデックを必要とするパッケージも少なくな
い。

今後はごく簡単なパラメータのインプットで監査ソフトウェアが使用で
きるように、マン・マシン・インタフェースの向上が望まれる。いわゆ
るユーザ・フレンドリーなシステムへの期待である。FOCUS, Easy-
trieve, MAPPER などの第 4 世代言語や簡易言語的思想を採り入れて、
日本語でインプットでき、日本語でアウトプットが可能な会話型ソフトウ
ェアが望まれている。

さらに、データの検索、サンプリング、簡単なロジカル・チェック等は
できても、複雑なロジカル・チェックとなると困難なものが多い。これか
らは、データベースも多様化し、複雑化していくだけに、昨今は特に資源
管理 (IRM) の重要性が叫ばれている。従来からのブラックボックス・
テスト (機能テスト) に加えて、ホワイトボックス・テスト (構造テスト)
の必要性が高まっているゆえんでもあり、そのためには、マッピング法な
どの支援ツールのレベルアップが必要となつてこよう。

また、入力データの正確性を確保するために、入力データのプログラム
・チェック機能解析ツールや、アプリケーション・ソフトウェアに対する
モニタリング機能組込みツールの開発も必要となつてこよう。

(3) 小型機でも使用可能なソフトウェア

システム監査支援ツールが求める使用リソースの制約の緩和、あるいは
マシン・タイムや使用リソースの負荷の軽減などは現実のシステム監査環
境での共通的要求である。

特に、既往の監査支援ツールは、そのほとんどが大・中型以上の汎用コ
ンピュータ・システムを前提としているといってもよい。最近では、パソコ
ン、オフコンといえども、飛躍的に性能が向上してきているだけに、スタ
ンドアロン型でも、また遠隔地のメインフレームの端末機としてでも、パ
ソコンやオフコン・ベースで使える監査支援ツールの開発が望まれている。
一部では PC-FOCUS や Easytrieve PC など、すでにパソコン・ベ
ースの汎用ソフトウェアも監査支援ツールとして使用可能になっているが、
今後さらに各種のパソコン・ベースの汎用監査ソフトウェアの開発を期待
したい。

(4) 監査パッケージの移植性の向上

昨今の大規模ユーザは、国産機、外国機を問わず、複数の異機種コンピュータを導入し、適用業務別に機種を使い分けているケースが多い。また、外部監査人も、クライアントが複数にわたる場合、特定の監査支援ツールが特定の機種には使えても、他機種には使えないという現象が出現する。OSが異なっても、ある種の中ドル・ソフトウェアなどを介することによって、他機種にも使いたいという要望が高まっている。

これはまた、OSのバージョン・アップやシステム構成の変化にも即応できるツール利用の継続性への要求にも通じるものである。

(5) 業種別監査支援ツールの開発

中央省庁、地方自治体、保険業、証券業、銀行、医療機関等でも、監査のニーズが高まっているにもかかわらず、現状ではある程度業種別に特化した監査支援ツールはほとんどないといってもよい。汎用のパッケージ・プログラムが、業務別ソフトウェア開発から業種別のソフトウェア開発に移っているように、これからはニーズの多い業界向けの特殊事情を考慮した監査支援ツールの開発も必要となつてこよう。

前述のように、汎用化すればするほど、JCLやパラメータのセットなどに手数を要するため、業種オリエンテッドなソフトウェアの登場が望まれている。

(6) リスク分析支援ツール

リスク分析については、定量化のメソドロジーが必ずしも確立しているとはいいがたく、またリスクの性質によっては、定量化がそもそも困難なものもある。

しかしながら、現状は、リスク分析の大半がかなり定性的、属人的に行われており、今後の研究開発によって定量化できる分野も十分残っていると思われる。リスク対策には必ず投資をとらなうが、その投資額の妥当性はできるだけ客観的な評価を必要とするだけに、まずリスク定量化のメソドロジーの開発、次いで必要なリスク分析支援ツールの開発が望まれる。例えば、各種のリスク・ファクターを入力すれば、定量化とリスク分析のためのチェック・リストがアウトプットされるプログラムなどがその一例であろう。米国でも、すでにist/RAMP(Risk Analysis and Management Program)などが開発されている。

(7) 効率性監査支援ツール

コンピュータ・システムの開発・保守効率の測定については、従来からいくつかの試みがなされてきた。例えば、積上法、ファー・アンド・ザゴルスキー・モデルに見られるようなパラメトリック法、アルブレヒトの提案するファンクション・ポイント法などがある。

また、コンピュータ・システムの運用効率については、ベンチ・マーク法、ギブソン・ミックスに代表されるような命令ミックス法、カーネル法、合成プログラム法、ソフトウェアやハードウェアによるモニタリング法などが開発されている。

しかしながら、コンピュータのバックグラウンドが少ない監査人にとっては、いずれもこれらの手法を使いこなすのは容易ではない。システムの効率性の測定は、システム監査の3大要件の1つであるだけに、効率性に焦点を当てた監査支援ツールの開発も今後必要になってこよう。

(8) セキュリティ・チェック支援ツール

システム監査では、信頼性・効率性ととともに、安全性チェックが大きな柱である。その意味からも、セキュリティのスター性チェック(178ページ参照)など情報フロー分析ツールや、データベース推論制御技術の開発も必要となってこよう。残念ながら、現状ではこの種の分野の開発はまだまだ遅れているといわざるをえない。

(9) オーディット・オートメーションのための監査支援ツール

一部の公認会計士事務所などではすでに実用化しているが、監査計画作成支援システム、監査調書作成支援システム、スケジュール管理システム、クライアント・データベース、技術計算支援システム、確認状作成管理システム、データ保護システムなど、監査業務のOA化のための支援ツールも重要である。只今現在では、そのほとんどが特定の外部監査人のための専用プログラムで、内部監査人が利用できるような形で提供されているものはきわめて少ない。

(10) システム監査支援用エキスパート・システム

エキスパート・システムが実用化されるためには、エキスパートの思考プロセスが明確化されていることが前提である。システム監査人の思考過程は必ずしも構造化(structured)されているとはいいいがたいが、逆にまたその全部がブラックボックスにとどまっているわけではない。今やAI

技術の導入によって、監査情報を分析するのに必要な専門知識やアプリケーションの専門知識の知識ベース化への取り組みが始められてもよい時期に入ったのではないと思われる。例えば、出力データから入力データを推論するツールなどは診断エキスパート・システムの一例といえよう。

6.1.2 監査実施上の課題

(1) システム監査支援ツールの分類

システム監査に使用する各種のツールはその性格から二つに分類出来る。第一はシステム監査人のみが使用するものであり、第二は社内のシステム部門が使用するものをシステム監査人が一緒に使用するものである。これらはシステム監査人の立場から見ると夫々に問題を持っているので、順次検討してみる。

(2) 多 様 性

システム監査の内容はその企業のシステムが抱えている問題に応じて、又、時期によっても極めて多様である。システム監査は、公認会計士による財務諸表監査のように一定の目的と範囲を持つものではなく、全般的にシステムの開発及び運用の管理水準の向上を目指す全ての監査活動であるから、監査人の問題意識によって極めて多様な領域に及び、それは時とともに変化して行くという基本的性格を持っている。従って使用されるツールも監査のテーマに応じて適切なものが選択されるので、勢い極めて多様なものとなる。極言すれば、システムの持つ全ての機能は時に応じてどれでもシステム監査のツールとして利用出来ると言っても良い。

(3) システム監査人専用ツールの問題点

代表的なものに『汎用監査ソフトウェア』があり、システムに組み込まれた監査用モニタ・ルーチンもこれに当る。又、あるツールがたとえシステム一般に利用可能なものであっても、社内でシステム監査人のみが使用する目的で導入されたのであれば、この分類で考えなければならない。例えば、統計的サンプリングのパッケージをシステム監査人主導で導入し、経営管理目的等で他の人が使用することがない場合がこれに該当する。

重要なことはこれらのパッケージが、現在は勿論、長期的にも自社のシステム環境に適合していることである。会計監査事務所によって開発された初期の汎用監査ソフトウェアは、パンチカードと磁気テープに記録され

た順次編成ファイルのデータを前提としたものであった。現在の大規模なコンピュータ・センターでは、パンチカードを使用することは稀である。カード・リーダは緊急用に保存されてはいるが、機械室の片隅でカバーを被ったまま眠っていると言うのが実状である。このような環境でラン毎にカードでJCLを流さなければならないジョブは現実的とは言えない。たとえこれをTSS端末機からコマンドを入力出来るように改良したとしても、カード・イメージの一方的コマンドの連続では、会話型のシステムに親しんでいる現在のシステム担当者に喜ばれるものではあり得ない。特定の監査人が極く限定的に使用するのであればともかく、重要な武器として導入するのであれば、問題を残していると言わなければならない。

特にこのようなパッケージはOS等のリリース・アップに際して共通のサポートを得ることが出来ず、取り残される危険がある。そして、そのようなメンテナンス負担が全て監査人自身にかかって来るようであれば、そのための特別な技術知識を集団として維持していなければならないと言う大きな荷物を背負い込むことになる。

このような問題点はシステム監査支援ツールに限ったことではなく、購入パッケージ全般の条件として大切な点である。システム監査人の立場は、社内にこのような非効率なものがないような環境保全を推進することにある。社内のユーザ部門のためにオープン・システムが稼動しているとして、特定のユーザのためのパッケージやユーティリティが、システムの本流でないために、メーカーSEや社内テクニカル・サポートによる十分な支援サービスが受けられないとしたら問題である。

システム監査人の業務は、やはり特殊であり、現在の巨大化したシステム部門全体から見れば小さなものであろう。それだけに今後の発展のためにも特別扱いではなく、システムの中核から外れないで、十分なシステム技術の保護を受けられるような位置を守って行く必要が大きい。

結論として、汎用監査ソフトウェアのようなものでも、たまたまどこかで開発されたものを適宜使用すると言った位置付けで監査人が利用するのではなく、様々なシステム環境の中で最適なものとして維持・発展して行く保守を保証されたソフトウェア商品になっていないと普及しないと思われる。

(4) 開発支援ツール・ユーティリティ活用の問題点

① 監査人のスキル

実務上最も残念に思うことは、監査人に多くの開発支援ツールやユーティリティを活用するスキルが十分でないことである。これは単にシステム監査人の勉強、努力が不足しているからではなく、対象があまりにも多く、複雑だからである。例えば、SMFのように多様な機能を持つツール体系を目的に応じて使いこなすには、常時十分な勉強と訓練が必要であり、高度な専門職でなければ不可能である。比較的簡単なツールでも、ラインで効率良く使われているのと同様に使いこなすには相当な経験が必要であるが、システム監査の内容は極めて広いため、個々のツールに習熟するに至らず、次のテーマに挑戦せざるを得ないのが実態である。このため現在では、個々にラインのサポートを得つつ、何とか目的を達している状況である。これから脱却する道は唯一つ、システム監査人の数が増えて、多様な技能の持ち主の集団となる程に大きくならなければ解決されないように思える。

システム監査人が新しいツールを習得しようとする場合、システム部門の新人教育や、新機能導入教育の機会を掴めるのは最も幸せなケースである。しかし、多くの開発ツールやユーティリティは小集団のプログラマ・チーム内で先輩から後輩に伝達されることが多い。システム監査人は未だそのような集団に成長していないし、そう出来るような似た業務の繰返しでもない。分り難いマニュアルを独習して、余り嫌われないように気を使いながらラインの専門家に質問しながらスキルを磨くことが必要なのである。

必要なスキルについて監査人に適した教育を受けられるような規模になるのは何時の日であろうか。

② システム監査の定型化、定例化

システム監査支援ツールを使いこなす一つの道は、業務内容の定型化と定例化を進めることである。システム監査として取り上げるテーマと手法が定まり、一定サイクルで実施することとするならば、その範囲で選択したツールは反復使用されることとなる。そうなればツールに対する需要も一定し、監査人のための改良や教育もやり易くなる。そして使用スキルも向上することとなる。

先般来、システム監査の内容は多様であると主張して来たが、これは裏返して言えば、未だシステム監査が未熟であり、やるべきことを全てやっているわけではないということである。理想的には、この未だ征服されていない領域を十分に把握整理して、監査活動の80～90%を定型化、定例化し、残りの10～20%程度を新しい変化に対応するものとして流動性を持ってマネージ出来るようになれば、それに応じたツールも開発、流通が進むことになるだろう。

ここで一つの監査活動の例示を行う。

- ・システム監査人は不当なプログラムの改ざんを発見、防止するために、毎月1回、前月中に改訂が行われたプログラムを監査することとする。
- ・開発部門から前月中に改訂した『プログラムのドキュメンテーション』を提出させる。
- ・運用部門から前月中に受付けた『プログラムライブラリ変更依頼書』を提出させる。
- ・システム監査人はライブラリ管理システムにアクセスして、前月に改訂登録された『プログラムのリスト』を出力する。
- ・以上の3者は全て一致することを突合確認する。
- ・システム監査人は新プログラムのソースとロードモジュールが同一であることを確認し、更に旧プログラムと新プログラムを比較し、相違する箇所はドキュメンテーション上で承認済の改訂箇所のみであることを確認する。
- ・そのツールはプログラム・コンペアである。現在はソース・コンペアが一般的である。理想的にはロードモジュールに対して直接コンペアをかけることが出来るように改良されることが望まれる。

③ システム監査業務のライン化

開発管理ラインと同様なツールをシステム監査人が使用して、有効な監査活動を定着させた場合、次のステップはその業務のライン化である。前記例示のような改訂プログラム管理は、システム監査として一定の役割を果たし、その有効性が検証されたならば、システム管理ラインに移管することができる。即ち、監査部門による定例的レビューは、管理担当による定例的チェックに委ねることによって、継続的、^{しつこい}悉皆チェックとして定着することができる。このためにはツールを使用する環境とそ

の効率が定例業務として耐え得るに十分なものでなければならない。

システム監査ツールにも、日常業務と同様な、使い易さと効率が保証されていないと本来的発展はあり得ない。

④ 高度、大規模なツールの共同利用

システム監査ではシステム管理ラインとツールの共同利用を図ることが出来る。と言うよりも、共同利用する以外には道がないような性格のツールが有ると言うのが正しいであろう。大規模な効率測定システムや開発管理システムがこれに当たる。例えば、開発管理システムでは、データの入力が開発担当者又は管理ラインで行われなければならない以上、監査人が利用出来るレベルに達することは困難であろう。監査の必要性を主張しただけで企業として負担出来るものでないことは議論の余地もあるまい。

システム監査目的で主張出来ることは、負担をあまりかけない範囲で特定の入力項目設定を増加させる程度に止まるであろう。しかし、企業全体が求めている監査活動としては外すことが出来ないような重要問題であることが多いので、システム監査人は積極的に管理ラインと共同して、必要なデータの確保とシステムの活用を図って、管理水準向上に努めることになる。

(5) 経済的負担

システム監査がどの程度の予算を許されるかは、明確な基準がない。今後、実際の効果として認められる定量的効果と定性的効果を積上げて、自ら予算を獲得して行くことが必要であろうが、日本的経営の中でどこまで出来るかは心配である。こうした間接部門に現在許されるであろう範囲を想像するだけでも、諸ツールの利用に必要な導入コストとコンピュータ資源は極めて大きいように思われる。仮に監査人が独自に簡易言語と監査機能パッケージを導入しようとするれば、約1千万円を要するであろう。ツールとしてユーティリティを導入しようとしても、月額10万円単位のものもある。

漸進的向上を図る公として、極力システム管理ラインとの共通性を重視することが最善の道と思われる。

6.2 新しいシステム監査支援ツールの考え方

前節で述べた課題をふまえ、今後新たに研究開発すべき監査支援のためのツールや技法の考え方を以下にのべる。実現可能性に対する、より具体的な検討は今後の課題である。

6.2.1 情報処理システムの変化に対応するツール

前述のように従来のシステム監査支援ツールは情報処理技術の進歩や、情報処理システムの変化に充分追従して来ておらず、基礎的には監査の態様が図6-1のように変化すべきである。

以下に具体的な技術項目との対応をとりつつ幾つかの例示を行う。

(1) データベース

データベースの有用性を支える四つの要素C.A.R.P. (Controllable, Auditable, Recoverable, Protectable)のうち、現在なおAuditable即ち監査性に関してはまだ充分な対策がとられているとは言えない。

今　ま　で	今	これから
静　態　監　査	動　態　監　査	
データファイル	データベース	
断　片　的	連　続　的	
バ　ッ　チ	リアルタイム	
スタンドアロン・ユーティリティ	ビルトイン	

図6-1 監査の態様の変遷

従来のデータ監査用のツールは、主としてファイルを対象としており、したがってデータベースを一旦ファイルに分解し、再編成を行ってツールを適用するという事が止むを得ず行われることもあった。しがしながら、これではスタティックなテストに止まり、特にランダム・アクセスの動態を監査することは不可能である。

データベースの監査の目標は図6-2の中央の4項目と考えられ、またそのために監査人が行う業務は、周囲に示す6項目の視点に要約すること

が出来る。このような目的を達成するための最善の手段は、DBMS自体に監査機能を内蔵させることであろう。

例えば、監査人が指定するデータやトランザクションが、処理の過程に於て自動的に監査用ファイルに記録されるのみでなく、オンライン端末によってその内容が自由にアクセスされ、また第4世代言語の拡張として、監査用機能が設けられることも必要かも知れない。

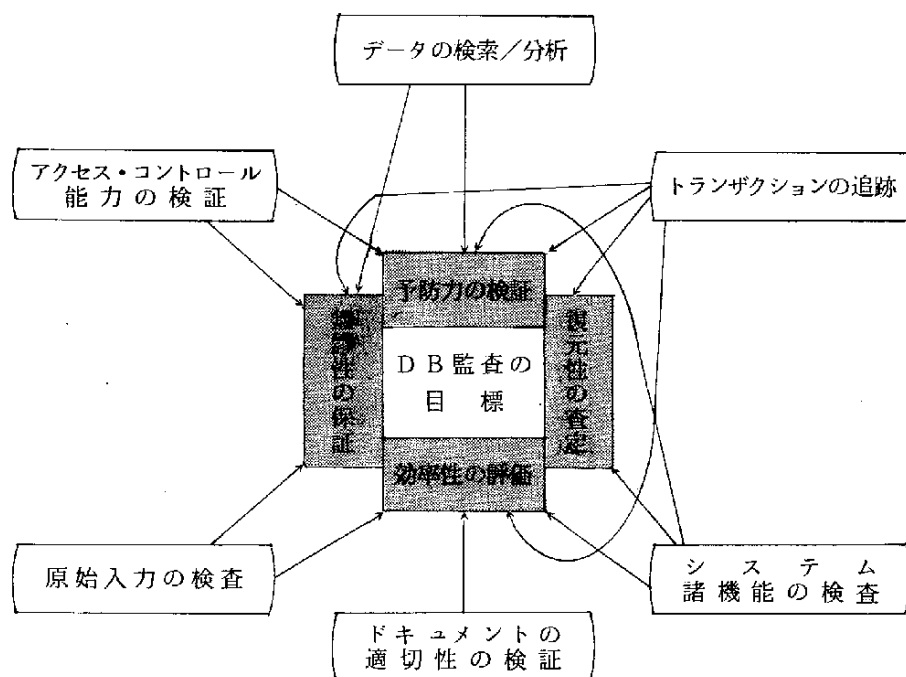


図6-2 データベース監査の視点

(2) ネットワーク

通信と情報処理の融合によって、組織内、組織間あるいはより共用性の高い公共網など各種のネットワーク化が急速に普及して来た。その規模も国際レベル、全国レベル、地域レベルなどから、さらに1組織内や1構内のローカル・エリア・ネットワーク(LAN)まで大小様々な多様化がみられる。また多数のコンピュータを接続したコンピュータ・ネットワーク・システムに於ては、システムの総合的な有用性や効率性を向上させるため、コンピュータ側とネットワーク側の機能分担に於て、かなりの付加価値機能をネットワークに整備することが求められ、情報処理システム全体

におけるネットワークのウェイトは、さらに増加の方向にある。

そこで当然ながらシステム監査の対象も、単にホスト・コンピュータ内の処理のみに限定することが不可能となり、ネットワークの処理機能自体の監査が必要不可欠となりつつある。

ネットワークを対象とした監査視点としては、例えば以下のような機能が対象となろう。

【ネットワーク自体の機能】

- ① 通信制御プロセッサ，フロント・エンド・プロセッサ，端末制御用機器等における通信制御処理機能の信頼性と安全性
- ② 通信プロトコルの標準や基準に対する準拠性
- ③ 通信ネットワークの障害対策機能の実効能力
- ④ 通信ネットワークのセキュリティ機能（コールバック，デジタル署名，暗号化，相手確認，閉域接続，不正介入防止等）の正当な稼働
- ⑤ ネットワーク接続におけるゲートウェイ機能の正当性
- ⑥ セキュリティ・レベルの異なるネットワーク間接続のセキュリティ管理機能
- ⑦ 接続ホストのセキュリティ・レベルの差異に対するコンピュータ・ネットワークのセキュリティ管理機能

【ネットワーク機能のウェイトが大きい情報処理システム】

- ① 電子取引，EFTS（Electronic Funds Transfer System）などの信頼性，安全性対策機能とその実効能力
- ② 分散処理システムのセキュリティ機能（ダウンラインローディング，分散プロセッサ内のソフトウェア・ファイル保護，不正入力防止機能等）の実効能力
- ③ O Aシステムのセキュリティ機能（ワードプロセッサ・ファイルの保護，LAN機能を含む各ワークステーションの本人確認，アクセス制御，暗号化等のセキュリティ機能）の実効能力
- ④ メーカーの遠隔保守時における主としてアクセス制御関連の安全性対策

以上のような種々のネットワーク関連機能を監査するには、少なくとも何等かの監査用端末あるいは監査用ワークステーションによってリアルタイムにモニタリングが行えるほか、モニタ結果のファイル化や、擬似トラ

フィックの発生やシミュレーション機能なども必要となろう。

(3) F A 関連システム

CAD/CAM/CAE (Computer Aided Design/Computer Aided Manufacturing / Computer Aided Engineering), ロボティックス, FMS (Flexible Manufacturing System) などを含むFA (Factory Automation) システムは, 情報処理の新しい分野の一つである。

OA が事務機器とコンピュータの合体であるのに対し, FA は機械とエレクトロニクスの合体であり, 信頼性, 安全性の意味も, さらに物理的な要素にまで拡大される。

システム監査という観点からは, 厳密に何処までをカバーすべきかの議論もあるところだが, 少なくともコンピュータのハードウェア, ソフトウェアが関わる部分は, 当然対象となる。

監査の手段はやはり, FA-LAN に接続された監査用端末やワークステーションによって行われる可能性が高いが, マイコン内蔵的な機能が多いだけに, 具体的な監査内容や監査方法に関してはなおかなり検討の余地が多い。

いずれにしても増々大規模化するプロセス制御等のシステムと類似するものとして, 従来とはかなり異なる発想と観点によるシステム監査技術が必要不可欠となろう。

(4) 人工知能応用システム

現在多様なアプリケーションに対し, 人工知能 (AI) 的アプローチがさかんである。

システム監査と AI 技術との関わりには二つの面がある。一つはシステム監査支援ツール自体への AI 的手法の導入である。もう一つは各種の AI システムに対するシステム監査技法の開発であり, そのためのツールの検討である。

この両者は将来的な具体化の形態では, かなりの共通性が得られるかも知れないが, 当面の検討はそれぞれの目的に対して別個に行う必要がある。

① システム監査支援ツールへの AI 手法の導入

多くの AI 技術の中で, 比較的早期にシステム監査に応用し得るのはやはりエキスパート・システムあるいは知識ベース的機能であろう。自然言語理解や質問応答機能も, システム監査のマンマシン・インタフェ

ースの向上に大きく寄与すると思われるが、これらは他のアプリケーションにおける要求との共通点も多い。

人工知能システム指向のツールの考え方の例を以下に示す。

イ. 出力データから入力データを推論するエキスパート・システム

システム監査側あるいは開発者側が通常行うアクションは、入力データから出力データを推定することであるが、これは仕様書やプログラムの解析という従来型の方法によって可能である。しかしその逆の、出力結果から入力データを推定するという行為は一般にかなり難しい。数学的には

$$Y = f(X) \quad \text{ここで, } Y : \text{出力結果}$$

$$X : \text{入力データ}$$

に対し,

$$X = f^{-1}(Y)$$

という逆関数 f^{-1} を求め、入力データ X を推定する仕事である。

これは、ある装置に何等かの障害が発生した場合、その症状から故障の原因を推論する診断エキスパート・システムと相通ずるものがある。

このようなツールは、一つのソフトウェアやプログラム全体を単位として対象とすることもできるが、現実には、むしろその部分、部分に対し適用することで、より効用が高まるとと思われる。当然ながら、プログラム・エラーの原因発見用ツールとしても有効であろう。

ロ. システム監査支援の知識ベースシステム

システム監査を実施するにあたっての種々の専門知識やノウハウを可能な限り体系的に知識ベース化し、自然言語的に検索し得るシステムである。知識ベースには例えば

- ・システム監査のインタビュー技法
- ・質問内容
- ・監査上のチェック・ポイント
- ・目的別の技法やツール
- ・ツールの使用法
- ・報告書の作成方法

等が含まれる。

ハ. アプリケーション・システム理解のためのシステム監査支援ツール

一般に、あるアプリケーション・システムを監査するには、その内容に対する知識がまず求められるが、数多くのアプリケーションが存在する場合は必ずしも容易ではない。

アプリケーションの内容理解の手段は、要求仕様書、機能仕様書、各種設計書等であろう。これらの文書をコンピュータによって分析可能にするには、何等かの標準記述法が存在する必要があるが、開発技法との関連によってそれが可能となるなら、アプリケーションの内容についての知識をコンピュータ化し、監査人を支援することができる。

また汎用的アプリケーション（例えば会計システム、給与システム、販売・在庫管理システム等）の共通的なシステム監査上の知識と併用することによって、異常値チェックなどを始め、定型的部分のシステム監査の自動化の可能性も生れてくるであろう。

ニ. 監査情報分析用エキスパート・システム

現在のシステム監査支援ツールは、図6-3のⅠ，Ⅱ，Ⅲを支援す

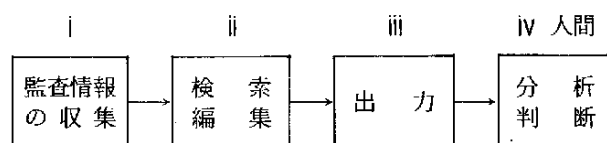


図6-3 監査情報分析

るものが殆んどであり、Ⅳの分析・判断は人間が行っている。そこでこの分析・判断業務を支援するエキスパート・システムというものが考えられる。この種のシステムでは、推論過程の説明機能が必要不可欠であり、これによって不足しているデータや知識を継続的に更新することによりシステムのレベルアップがはかれる。

② AIシステムに対するシステム監査技法

AIの応用には、表6-1に示すような各種のシステムが存在する。

表 6 - 1 A I システムの例

自然言語処理システム
パターン認識システム
機械翻訳システム
質問応答システム
知能ロボット
知的 C A I システム
知的意思決定支援システム
自動プログラミング
エキスパート・システム

これらの各々のシステムの目的や特性に応じた個別の監査内容や手段を検討することが必要であろう。ただ現在、これらの殆んどのシステムはまだ研究中あるいは試作段階にありシステム監査の対象とは認識されていないのが実情である。したがってその技法やツールを云々するのはあまりにも時期尚早と判断されるであろう。しかしながら、比較的早期にその実用化が期待されているエキスパート・システムに関しては、その急速な流行の実態からも、何等かのシステム監査的アプローチを準備しておくことは無駄ではあるまい。

現在エキスパート・システムは伝統的な A I の概念から一步踏み出して、推論機構と知識ベースという 2 要素によってアプリケーションを構築するという、一種のソフトウェア開発技法とみなされるケースが増えて来た。即ち、必ずしも明確なアルゴリズムや処理ロジックが簡単には得られないアプリケーションを対象とした場合、大変便利な技法としてエキスパート・システムの的アプローチがとられることとなる。ある意味では広義のシミュレーションとみなすこともできる。

しかしながらこのような段階を過ぎて、種々の応用分野における、診断、制御、設計などを始め、極めて重要な問題解決の手段として、エキスパート・システムが実用化された場合、その正当性や信ぴょう性に対する評価は必要不可欠となろう。

一般のソフトウェアに於ても、システム監査の基礎的要求は、そのブラックボックス性の打破にあるわけであるが、A I システムに於ても

同様である。何等かの結論を導くのに、どのような知識が使われ、どのような推論が行われたか、という説明機能は極めて有効な機能となろう。これは、一般のアプリケーション・システムの中に、モニタリング的な監査機能を内蔵させるという要求と同等のものである。

また現在の多くのエキスパート・システムに於ては知識ベース内の矛盾した知識は、推論機能自体の働きによってその矛盾を指摘し得る自浄能力を持っているが、さらに、より高度の自己評価・改善を行う学習的機能をAIシステム自体に内蔵させることは可能となるであろう。そして将来の理想としては、単にAIシステムのみならず、一般のアプリケーション・システム自体にも、類似の機能が内蔵されることが望ましく、AIシステムの監査を手がかりとして、その実現への道がひらけるかもしれない。

6.2.2 ソフトウェア作成段階における監査ツール

ソフトウェアの信頼性や安全性を監査するには、完成したソフトウェアを対象とするだけでは全く不十分であり、ソフトウェアの製造工程に立入った形での監査が必要不可欠である。その具体化には幾つかの手段が考えられる。

- ① 要求定義，基本設計，詳細設計，プログラム設計，プログラミング等の、各段階毎のレビュー時点で、それぞれの作業内容を監査するツール。
- ② ソフトウェアの設計時に、運用上の監査機能、例えば入力データのモニタリングと制御機能などをソフトウェア自体の機能の一部として組み込む。
- ③ 設計，プログラミング，テストの各段階でソフトウェア作成者が用いる作成支援システムやツールに監査人が使い易い機能を付加して改造する。

現在システム監査人は、特にシステム監査用に作られたものではなく、ソフトウェア作成者のためのユーティリティやテストツールを多数流用している。その是非はともかく、本質的には、やはりソフトウェア作成段階にかかわるシステム監査支援ツールは、ソフトウェア作成側が使用するツールとほぼ一致するのは当然であろう。例えば上述の①は、ソフトウェア開発管理におけるレビューやウォークスルーのツールとしても用いられ、②は、ソフトウェアの運用テストの一環としても効果があろう。ただ③のように、ソフト

ウェア作成側のツールをそのまま使用するよりも、監査人の使い易さを加味した種々の改良が加えられればより望ましい。

なお、この問題に関する詳細は、“第4章 情報システム開発技法とシステム監査”で述べたが、具体的ツール例のイメージを以下に三種類付記する。

(1) マッピング法支援ツール

ソフトウェア作成側でも用いているカバレッジ・テストは、プログラム構造内の全ステップを対象に、その何%がテスト済みであるかを計測するツールであり、ホワイトボックス・テストとも呼ばれている。

システム監査側に於てもこのツールを使用することは可能であり、また効果もある。ただその際、ソフトウェア作成側が使用したテストデータに関する情報が入手できれば、その作業はより効率的である。しかし、それが存在しない場合は、一通りのテストデータをシステム監査側で準備せねばならない。また、カバレッジを上昇させるにつれて、あるいは、テストしたい目的ステートメントを特定する要求が強い場合などにはプログラム記述書の詳細にわたる分析が必要となり、目的に合うテストデータを作成するのは簡単ではない。

そこで、このツールは、プログラム中のテスト標的ステートメント（あるいはステートメント群）を通過するテストデータを自動的に発生させるものである。

なお、本ツールの実現のためには、下記の機能の統合が必要である。

- ① プログラムの静的および動的解析
- ② テストデータ・ジェネレーション
- ③ テスト・カバレッジの測定

(2) ソフトウェア保守情報記録システム

ソフトウェアの保守作業は、現在のソフトウェア分野の課題の一つである。全てのソフトウェア作成作業の6割を越えるとともに、コスト的にも同程度にのぼると問題視されている。また、往々にして保守作業にともなう新たな誤りの混入の比率も高い。

一方、セキュリティの観点からすると、プログラムの不正な更新の脅威は大きく、その早期の発見手段が求められる。本ツールは、これらの課題に対処するための監査支援ツールの一種である。

本ツールでは、ソフトウェアの開発・更新環境を固定し、定められた方法以外の更新を禁止する。ソース・レベルの更新にともないドキュメントも自動的に更新され、すべての記録（更新者、日時、更新内容）が管理される。更新記録は監査人によって会話型で検索され、更新内容はわかり易く解説付で表示される。

本ツールの利点は、以下の通りである。

- ① 更新の正当性の監査を容易にする
- ② パッチによる不正更新が禁止される
- ③ ドキュメントが常に最新状態を保つことができる

また、本ツールの実現には、下記のようなソフトウェア開発環境が前提となる。

- ① WS（または端末）による会話型の開発環境
 - ② ソース入力または設計記述言語によるプログラム・ドキュメントの自動作成機能
 - ③ 保護機能を持つファイル管理機能
- (3) 入力データチェック機能解析ツール

各種アプリケーション・ソフトウェアに於いて、入力データの誤りを可能な限り事前に検出し、処理の正確性を確保することは必要不可欠である。

現実には、パッチ処理に於いては、独立したデータチェックのランを予め行い、クリーンデータを作成してから本番処理に入ることが多い。一方、リアルタイム処理の場合は、アプリケーション側で、可能な範囲の物理的・論理的チェックを行いつつデータを入力しながら処理を進める。

本ツールは、これらの入力データチェック機能の妥当性を監査するためのツールであり、プログラム中のチェック処理を追跡し、内容を分析し明示することにより監査を支援する。またこのツールによって、プログラム内のリミットチェックの判断条件の不正変更なども検出することが可能となる。

本ツール実現上の手がかりとしては、DD/DSとの連動によって、そのデータ項目に対する妥当性検査基準、クロス・リファレンス機能、メタデータ生成機能などを併用することが考えられる。

6.2.3 システム監査業務の作業環境の改善

システム監査支援ツールとは、いうまでもなくシステム監査業務を支援するものであるが、支援の手段として、特に操作性、使い易さ、効率性などを含め監査業務の作業環境の改善に重点を置いたツールというものが考えられる。

(1) 統合的に使用できるツール

例えば、要求仕様定義から設計、プログラミング、テスト、保守という一連のソフトウェア作成作業の過程におけるシステム監査を行う場合、各段階毎に別々のツールが散在するよりは、一貫した思想に基づき、共通的なマン・マシン・インタフェースを持つツールが存在することがより望ましい。

例えば監査用コマンド、要求指示のフォーマット、メッセージやヘルプ機能の統一性など、同一の知識や慣れが、そのまま活用し得ることは、操作性の上からも極めて有効である。

(2) 資源管理機能の利用

ツール側に資源管理的機能を持たせることにより、監査作業の効率化をはかることができる。例えば以下のようなアプローチが考えられる。

① マクロのカatalog機能に相当する能力を持たせ、使用の都度コマンド列を入力するような手間を省く。またそれらの手法をさらにレベルアップし、定型的監査プロセスの知識ベース化が可能となれば、より望ましい。

② DD/DSは組織内におけるデータ資源を管理するシステムであり、最近では、ソフトウェアの開発・保守・運用にかかわるデータの一元的コントロールにも利用されている。

一方、この機能はシステム監査にも有効であると見なされるようになった。例えば、監査業務における一連のテストデータのストックや、その再利用、回帰テスト等を支援する強力なツールになり得るであろう。また6.2.2で述べたソフトウェア作成業務過程とのリンクの一環として、ともにDD/DSをベースとして作業を行うことにより、いずれも生産性、効率上の利点が得られるばかりでなくソフトウェア作成とシステム監査のコントロール手法上の一貫性をはかることができる。

(3) システム・ソフトウェアとツールの関係

OSを含むシステム・ソフトウェアとシステム監査支援ツールとの関わり合いには二面性がある。

一つはシステム・ソフトウェア自体をシステム監査の対象と考えた場合のツールである。もう一つは、ツール自体をシステム・ソフトウェア・レベルの機能として持たせることの議論である。

① システム・ソフトウェアとその監査

システム・ソフトウェアとは以下のものを含む。

- a) 制御プログラム(狭義のOS)
- b) 言語処理プログラム
- c) 通信制御プログラム
- d) データベース管理システム(DBMS)
- e) セキュリティ・ソフトウェア
- f) 運用支援ソフトウェア
- g) ユーティリティ(サービス・プログラム)

これらのシステム・ソフトウェアは少なくともコンピュータ・システムの安全性・信頼性という観点からは中核的存在であり、その安定なくしては、システムの健全な運用はあり得ない。しかしながらこれらのソフトウェアは通常コンピュータ・メーカーやソフトウェア・メーカーによって製造され、ユーザ側がその設計・製造・テスト・保守の段階を監査することは難しい。したがって、一般ユーザは、その製品としての質はメーカー側に於て一応は保証されているものとして、運用上においてユーザ側の関る面のみを監査の対象とするのが普通である。

② 現在のシステム・ソフトウェア関連の監査支援ツール例

現在、システム・ソフトウェアの運用時の監査支援ツールの例としては以下のようなものがある。

イ. マルチプログラミング環境におけるプログラム実行コントロールの監査支援ツール

IBMのDOS/POWERは、DOS(Disk Operating System)における入出力の自動ステージングや優先順位スケジューリング等をユーザに提供するSPOOLing(Simultaneous Peripheral Operation On-Line)プログラムである。

DOS/POWERアナライザ(C&L社製)は、DOS/POWERによって作成される、その膨大な活動記録を分析し、レポートを作成するプログラムである。そしてこれによって以下のような点の監査情報を得ることができる。

- I. ジョブ・スケジューリングの妥当性のチェック(含、優先度設定の妥当性)
 - II. 各プログラムが必ず承認された予定表に従って実行されているか否かのチェック
 - III. オペレータの介入の妥当性のチェック
 - IV. ジョブ名、プログラム名の決め方のルールが遵守されているか否かのチェック
 - V. 各プログラムのスルーポイントがユーザ側の要件を満たしているか否かのチェック
 - VI. アクセスされたファイルの妥当性のチェック
- ロ. リンク・ライブラリ分析パッケージ

プログラムの変更は本来ソース・プログラム・レベルでのみ行われるべきであるが、“ZAP”によってオブジェクト・レベルや、ロード・モジュール(実行可能レベル)変更を行うことも可能である。しかしながらZAPによる変更はコンパイルおよびリンケージ・エディタによって行われるコントロール機能をバイパスすることが出来るため、その記録も残らず監査上のコントロールが難しい。

SLAPS(Special Linklib Analysis Program Series; C&L社製)はコンパイル、リンケージ・エディット等のデータに基づき、プログラムの作成・保守の記録をレポート化する機能を持つと同時に、ZAPの実行日や対象プログラム名等、ZAPが行われた記録もレポート化することが出来る。

これによってシステム・ソフトウェア、アプリケーション・ソフトウェアを含む、殆んどどのプログラムに対する保守や変更がとらえられ、監査上有効な手がかりが得られる。

ハ. SMFアナライザ

SMF (System Management Facility) は最近の中型以上のOSに備えられている機能の一つで、ハード、ソフトにわたるコンピュータ・システム資源の利用にともなう課金情報の記録、およびシステムのパフォーマンス・データの収録をおこなう制御プログラム機能の一種である。

SMFの記録ファイルは、本来のこのような目的に加えて、システム監査情報としても有効であり、SMFアナライザによって、以下のような監査上のチェックを行うことができる。

- Ⅰ. 実行されたジョブ名、プログラム名等から、予定され、承認された業務のみが行われているか否かのチェック
- Ⅱ. アクセスされたファイル名と、行われた処理(削除・変更等)、その利用者名のリスト等から、アクセス・コントロールが正当に働いているか否かのチェック
- Ⅲ. TSSやネットワークのアクセス時間、パスワードの誤りなどの記録から正当な利用が行われているか否かのチェック
- Ⅳ. アクセス制御プログラム、DBMSなどの使用記録から、それらの正当な利用状況のチェック

ニ. プログラム比較監査支援ツール

プログラムの同一性の検査は、不当な変更の有無を調べる目的からも重要である。

プログラム比較監査支援ツールは以下のようなケースの利用が可能である。

Ⅰ. ソース・プログラム同士の比較

ソース・プログラムに対する変更を発見する場合

Ⅱ. ロード・モジュール同士の比較

例えばテスト済みのプログラムや本番プログラムに変更が加えられたことを発見する場合

Ⅲ. オブジェクト・モジュールとソース・プログラムとの比較

オブジェクト・モジュールが、ソース・プログラムと一致するか否かのチェックであり、ソース・プログラムからオブジェクト・モジュールを生成し、オブジェクト・モジュール同士を比較する。

Ⅳ. 複数プログラムの比較

同一プログラムが複数のサイトで稼働している場合に、それらを
原版と比較し、不当な変更を発見する。

③ 新しいツールの考え方

以上述べたようなツールは、アプリケーションの監査支援ツールとしても用いられると同時に、システム・ソフトウェア自体の監査の可能性も持つツールである。

また当然ながらこれらの機能はシステム・ソフトウェア自体で持つべき機能でもある。

例えば、モニタリング関係の機能は制御プログラムの直接的コントロール下になければ、不可能であり、また比較ツールに関しても、ソース、オブジェクト、ロード・モジュール等すべてのモジュールはライブラリ管理の下にあり、同等のコントロールが及ぶレベルでのみ稼働可能である。

一方、現在の高度化された情報処理システムを対象とした場合、前述のようなツールに加えて、さらに以下のような機能・手段あるいは監査内容の拡大が必要であろう。

イ. モニタ機能の強化

i. モニタ能力の拡大

S MFに代表されるモニタリング・ツールは、現在の方法では、必要十分な詳細情報をとらえるには、処理効率的にも記録容量的にも極めて不十分である。そのため、本体とは別のロギング・プロセッサを併用したり、大容量の光ディスクを利用するなどの手段が検討される必要がある。

ii. 記録ファイルの保護

モニタリング・ファイル自体の保護が現在是不十分であり、オペレータ自身によって、その内容が変更可能の場合すらある。当然ながら特定の管理責任者や限られたシステム監査担当者だけにアクセスが許されるという厳格な保護環境が必要不可欠である。

そのためには、課金用、パフォーマンス評価用等のデータとの混在を避ける、収録データの性格別に、特殊コードの併用等によって記録方式に変化を与える、あるいは特に保護を要するファイルや、

レコードを分離し暗号化するなどの手段が考えられる。

Ⅲ. 不正利用に関するモニタリングの強化

アクセス・コントロールにおける不正アクセスの記録，本人確認におけるパスワードの誤りなどの記録，ネットワークへの不正介入の記録など，不正利用の可能性が考えられるアクションに対する記録は，IDコード，氏名，パスワード，日時，端末番号などを始め，より詳細な記録をとり，可能な限り監査証拠としての時系列変化の把握が望まれる。

ロ. FA関連システムの監査機能

FA関連のコントロール機能や，プロセス制御関連機能は，マイクロ・プロセッサ内蔵やメカニカル機器との連動性も強く，また何よりもリアルタイム性が極めて重視されるため，それらの制御システムと一体化した形での監査機能の組み込みが必要であろう。(6.2.1(3)参照のこと。)

ハ. アプリケーションのモニタリング機能組み込みツール

アプリケーションに対するシステム監査支援機能をOSレベルで持つか，アプリケーション・レベルで持つか，あるいはその中間的形態で持つかについては種々の議論がある。

現在は汎用監査ソフトウェアなどを始め，システム監査支援ツールとして独立しているものは殆んどアプリケーションのレベルで存在している。

一方，今後は効率や操作性の点からも，これらをシステム・ソフトウェア・レベルに引き上げる考え方も必要となろう。

例えば各アプリケーションの運用にともなう，不正入力データチェックや，リミットチェックなどのモニタリング機能は，それぞれのアプリケーション・ソフトウェア設計時に，ともに設計し，予めシステムに内蔵することが求められて来た。

しかしながら，設計時にそのような手続きを行わなかったアプリケーションや，運用後さまざまな変更をモニタリング機能に加えたい場合なども多い。そこで，それらに対処するため共通的機能を抽出し，システム・ソフトウェアのレベルに整備することによって，モニタリング機能の活用が大幅に容易になり，アプリケーション・システムの

監査性が向上する。

このような機能の実現上のヒントとしては以下のような手段が考えられる。

- i. アプリケーションの設計言語あるいはプログラムの記述言語にモニタリング機能指示コマンド、指示変更コマンドなどの機能を付け加え、それに応じたプロセッサ機能を強化する。
- ii. 種々のモニタリング機能モジュールを作成し、モジュール・ライブラリを設け、構造エディタにそれらの連結機能を持たせる。
- iii. 対話型の支援環境によって、これらの機能が監査用端末やワークステーションによって容易に操作し得る機能をもたせる。

ニ. 情報フロー分析ツール

情報フロー分析 (Information Flow Analysis) とは、プログラム中の変数に注目し、情報 (あるいはデータ) がどの変数を通過するかを分析し、セキュリティの*性 (スター性と読む) に違反しているか否かを検査する手法のことである。*性とは、任意のセキュリティ・レベルを保持するアクセス主体が自分よりも下位のセキュリティ・レベルを保持するアクセス対象に対して書込みアクセス操作を禁止する (すなわち、no write down : 下へは書けない) という規則である。

情報フロー分析手法では、プログラムあるいはそのプログラムの形式的仕様書に現れる変数すべての間での情報フローを識別する。

情報フロー分析ツールに於ては、まずセキュリティ・ポリシーに従って、各変数にセキュリティ・レベルが割り当てられる。セキュリティ・レベルとしては、束 (lattice) がよく採用される。束は半順序集合で、次のような条件の時のみ情報のフローを許可する。即ち変数 X のセキュリティ・レベルを $\underline{X}$ で表すとする、変数 X から変数 Y への情報フローは、 $\underline{X} \leq \underline{Y}$ の場合、かつその場合にかぎって許可される。

この手法では、文 (statement) の型に応じて、情報フローを構文的に識別する。例えば、代入文

$A := B$

は、 B から A への情報フローを意味するし、条件文

if $B = 0$ then $A := 0$

も同じ意味となる。

本ツールの実現のヒントとしては、プログラム仕様書およびプログラム言語仕様に形式的記述を導入し、それにもとづき処理の流れに沿って情報のフローを分析することが考えられる。

なおこの種の情報フロー分析の研究は、米国のMitreおよびSRIなどでも行われている。

結

言

結 言

システムの信頼性や安全性の完全な監査が困難である理由は、ソフトウェアというものの完全性を実証し得ない現在の開発側自体の課題と同根のものである。

したがって、ソフトウェアの可視性を高め、その正確性、妥当性、完全性を厳密に評価し得る手法・手段の出現は、開発側にとって待望久しいものであると同時に監査側にとっても絶大な効果をもたらすこととなろう。

即ち、それぞれの立場上、視点の差異はあっても、システムの信頼性や安全性の監査における技法や技術は、基本的には開発側の要求と一致し、今後は、双方が共に協力してその技術開発を推進することが可能となろう。

このような観点から、次年度は、開発技法に関連づけた監査技法を中心に、さらに具体的な検討を続けることとしたい。

付 録 システム監査支援ツール・技法の実態調査票

付録。システム監査支援ツール・技法の実態調査票

システム監査支援ツール・技法
の実態調査調査票記入要領

1. A票は、システム監査に支援ツールを使用している場合に記入して
下さい。

A票一部に付き一つのツール・技法をご記入下さい。（用紙は3部用意しましたが不足の場合には、お手数ながらこの票をコピーしてご記入下さい。）

なお、ここでのツールとは、コンピュータで動作可能な物を指します。

2. B票は、現在システム監査支援ツールを使用していない場合に記入
して下さい。

また、既に使用している場合でも、今後新たなツールを使用する予定のある場合は、B票にもご記入下さい。

3. C票は、システム監査支援ツールの使用の有無に関わらずご記入下
さい。

4. A票、B票にある「ツール・技法」の種類については、同封しました別添の説明資料をご参考にして下さい。

5. 選択肢の有るものについては、該当するものに○印を付けて下さい。
（複数回答可）

II. ツール・技法の種類

- | | |
|-----------------------|--------------------|
| 1. 監査領域選択法 | 2. 得点法 |
| 3. 複数現場用監査ソフトウェア法 | 4. 汎用監査ソフトウェア法 |
| 5. トランザクション選択法 | 6. 組込み監査データ収集法 |
| 7. テスト・データ法 | 8. 基本事例システム評価法 |
| 9. 総合テスト法 | 10. 平行シミュレーション法 |
| 11. スナップショット法 | 12. マッピング法 |
| 13. トレース法 | 14. コード比較法 |
| 15. 統制流れ図法 | 16. 監査手引法 |
| 17. ジョブ会計データ分析法 | 18. 災害テスト法 |
| 19. システム開発時点統制ガイドライン法 | 20. システム開発ライフサイクル法 |
| 21. システム検収及び統制のグループ | 22. 導入後監査法 |
| 23. カバレッジ・テスト | 24. オンライン・モニタ |
| 25. その他〔 〕 | |

Ⅲ. ツール・技法の概要 (パンフレットや解説書が有れば、それを添付して頂いても結構です。)

1. 区分 (a. 購入、b. 自作、c. 改造)
2. 使用形態 (a. 会話形、b. バッチ形)
3. 被監査システム (a. オンライン、b. バッチ)
4. 機能概要
5. 入力情報
6. 出力情報
7. その他

IV. 監査対象

- | | |
|-------------|--------------------------------|
| 1. データ・ファイル | 2. システム (OS, DBMS, アプリケーション 等) |
| 3. プログラム | 4. 手続き・方針・標準 |
| 5. 人 (要員) | 6. その他 () |

V. 使用段階

- | | | |
|---------|---------|---------|
| 1. 企画段階 | 2. 開発段階 | 3. 運用段階 |
|---------|---------|---------|

VI. 監査目的とその監査でのツールの有効度合 (上、中、下) を記入して下さい

- | | |
|-------------------|-------------------|
| 1. 準拠性の監査 (上、中、下) | 2. 信頼性の監査 (上、中、下) |
| 3. 安全性の監査 (上、中、下) | 4. 効率性の監査 (上、中、下) |
| 5. 有効性の監査 (上、中、下) | 6. 採算性の監査 (上、中、下) |
| 7. その他 () | (上、中、下) |

VII. ツール・技法を使用した効果を、次の各々の面からご記入下さい

1. 監査作業の効率 (a. 大変良い、b. 良い、c. 変わらない)
【理由】

2. 監査結果の精度 (a. 大変良い、b. 良い、c. 変わらない、d. 良くない)
【理由】

3. 監査の範囲 (a. 広がった、b. 変わらない)
【理由】

4. 監査期間 (a. 短縮した、b. 変わらない、c. 長くなった)
【理由】

5. その他
【理由】

Ⅷ. ツール・技法の問題点を、次の各々の面からご記入下さい

1. 適用限界
2. ハードウェア環境整備
3. ソフトウェア環境整備
4. 使うための準備（インプット・データの準備等）
5. 操作の容易さ等
6. アウトプットの情報項目、質等
7. 被監査プロジェクトまたはシステムへの影響
8. その他（改善点、要望点等）

1. ツール・技法をシステム監査に使用していない理由は何ですか

1. システム監査の必要性をやっと認識した段階である
2. システム監査の方法を構築している段階である
3. システム監査をやっと実施し始めた段階である
4. ツール・技法にどのような物があるか知らない
5. ツール・技法が使用目的に合致しない
6. ツール・技法が使いづらい
7. 使用環境が整っていない
8. システム監査の体制が整っていない
9. その他〔 〕

II. 使って見たいツール・技法を選んで下さい

- | | |
|-----------------------------|--------------------|
| 1. 監査領域選択法 | 2. 得点法 |
| 3. 複数現場用監査ソフトウェア法 | 4. 汎用監査ソフトウェア法 |
| 5. トランザクション選択法 | 6. 組込み監査データ収集法 |
| 7. テスト・データ法 | 8. 基本事例システム評価法 |
| 9. 総合テスト法 | 10. 平行シミュレーション法 |
| 11. スナップショット法 | 12. マッピング法 |
| 13. トレース法 | 14. コード比較法 |
| 15. 統制流れ図法 | 16. 監査手引法 |
| 17. ジョブ会計データ分析法 | 18. 災害テスト法 |
| 19. システム開発時点統制ガイドライン法 | 20. システム開発ライフサイクル法 |
| 21. システム検収及び統制のグループ | 22. 導入後監査法 |
| 23. その他〔 〕 | |

Ⅲ. ツール・技法を使用するときの監査対象は何ですか

1. データ・ファイル
2. システム (OS, DBMS, アプリケーション 等)
3. プログラム
4. 手続き・方針・標準
5. 人 (要員)
6. その他 ()

IV. どのような段階にツール・技法を使う予定ですか

1. 企画段階 2. 開発段階 3. 運用段階

C 票

ツール使用の有無に関わらず記入

1. 情報処理システムの信頼性、安全性の監査のための支援ツール・技法についての意見又は要望をご記入下さい（現在、支援ツール・技法の使用の有無に関わらずご記入下さい。）

—— 禁 無 断 転 載 ——

昭和 62 年 3 月 発行

発行所 財団法人 日本情報処理開発協会
東京都港区芝公園 3-5-8
機械振興会館内
Tel 03 (432) 9372

印刷所 株式会社 三 州 社
東京都港区芝大門 1-1-21
Tel 03 (433) 1481

