

45-Z 001

ソフトウェア汎用化技術調査報告書

昭和 45 年 9 月



財団法人

日本情報処理開発センター

JIPDEC

45
Z001

この事業は日本自転車振興会の機械振興資金
による「昭和45年度情報処理に関する調査・
研究補助事業」のうち「ソフトウェア汎用化技
術の調査研究」の一部として実施したものであ
ります。

1

2

3

序 に 代 え て

電子計算機の利用およびプログラム開発の促進は、国民生活の向上および国民経済の健全な発展に寄与するところ大であり、その具体的な方策として、プログラムの流通を円滑にするとともに、情報処理サービス業の育成措置を講ずることが強く要請されております。

先に発表された産業構造審議会情報産業部会の答申にも見られるように、ソフトウェア技術の開発としてナショナル・プロジェクト等によるソフトウェアの開発促進およびソフトウェア開発資金の確保、ならびに情報産業の育成策として情報産業に対する助成、情報産業の秩序づけの措置、政府機関による情報化の推進、外資対策等が当面の重要課題となっております。

これらの諸問題を解決するため、今年3月情報処理振興事業協会等に関する法律が制定されましたが、この法律の中で情報処理の振興を図るため開発を特に促進する必要がある、かつ広く利用される種類のプログラムの開発を国の費用を投じて実施する等の情報処理の振興策が打ちだされています。

具体的な方策として、①開発を特に促進する必要がある、かつその開発成果が事業活動に広く用いられるプログラム（特定プログラム）のうち企業等が自から開発することが困難なものに対し積極的に開発する、②特定プログラムのうち企業等が開発したものについて対価を支払い、その権利を取得する、③以上により確保されたプログラムについて対価を得て普及する、等の業務を行なう政府関係機関の設立をみております。

このような現状に鑑み、当財団では「ソフトウェア汎用化技術調査委員会」を設置し、今後発足する汎用ソフトウェア開発機関のあるべき姿の方向づけを行なうとともに当面するソフトウェア流通に関する諸問題を解決するため

の指針を審議して参りましたが、ここにその成果として「ソフトウェア汎用化技術調査報告書」をとりまとめ諸方面の方々の参考に供することと致しました。

この際、本事業にご尽力を賜ったソフトウェア汎用化技術調査委員会の各委員、ならびにご支援を賜った関係各位に心より感謝の意を表しますとともに、本報告書が関係方面に活用され、わが国情報処理産業の発展の一助として寄与できれば望外の喜びと存じます。

昭和45年9月

財団法人 日本情報処理開発センター

会 長 難 波 捷 吾

ソフトウェア汎用化技術調査委員会

(敬称略・順不同)

委員長	野田克彦	電子技術総合研究所
委員	石原善太郎	三井東圧化学
	藤井純	日本ソフトウェア
	吹田昇	協栄計算センター
	河野忠義	日立製作所
	若曾根和之	日本貿易振興会
	石本幹郎	通商産業省
	広田慶次郎	通商産業省
	吉田剛	日本情報処理開発センター

事務局 (財) 日本情報処理開発センター技術部研究課

23

11

目 次

1 総 論	1
2 特定プログラムの類型化	6
2・1 プログラムの分類	6
2・2 特定プログラムについて	10
3 汎 用 化	13
3・1 汎用化へのアプローチ	13
3・2 標準化の推進	15
3・3 汎用性の評価	15
4 委託のための事前調査分析	17
4・1 事前調査の項目	17
4・2 事前調査の方法	18
4・3 事前調査の外注	18
4・4 事前調査と製作との関係	19
5 発 注 方 法	20
5・1 発 注 の 範 囲	20
5・2 発 注 方 法	21
5・3 発注方法上の問題点	21

6 対 価 の 決 定	23
6・1 内外における対価決定の現状.....	23
6・2 対価決定に対する考え方.....	25
6・3 考慮すべき対価決定の要素.....	26
6・4 対価算定方式の推奨パターン.....	28
6・5 対価決定の問題点.....	30
7 委託開発プログラムの進捗管理	33
7・1 進捗管理すべき範囲.....	33
7・2 進捗管理の内容.....	34
7・3 工程管理の推奨パターン.....	37
7・4 管理上の問題点、注意点.....	39
8 検査、検収方法	41
8・1 検査・検収の範囲.....	41
8・2 検査・検収の方法.....	41
8・3 メンテナンスの保証と期限.....	42
8・4 実施上の問題点.....	42
9 貸 付 け 方 法	44
9・1 貸付けの条件と手順.....	44
9・2 貸付けの形式.....	46
9・3 利用者に対するコンサルティング.....	49
9・4 運用上の問題点	51

10 貸与するプログラムのメンテナンスの問題.....	53
10・1 プログラムのメンテナンス.....	53
10・2 貸与するプログラムのメンテナンス契約.....	53
10・3 貸与するプログラムの機能改善 (revision up)	54
10・4 プログラムの機能強化 (version up).....	54



参考資料ー1 価格算定の諸方式	56
参考資料ー2 1年間に1人が生産するプログラムの大きさと種類	66
参考資料ー3 細分化された工程表.....	67
参考資料ー4 職種体系と要求される能力.....	68
参考資料ー5 パッケージド・ソフトウェアの販売状況.....	69
参考資料ー6 価格決定に対する考え方.....	72
参考資料ー7 パッケージド・プログラムの価格例.....	74
参考資料ー8 主要な汎用アプリケーション・プログラム.....	76

1. 総 論

この報告書は、汎用プログラムを委託開発し、これを貸付ける際に必要な諸事項を検討して、汎用ソフトウェアを委託開発し貸付ける機関（以下このような発注および貸付の機関を汎用ソフトウェア事業機関と呼称する）がかかる業務を実施する際の指針、あるいは実施にともなう行為の標準パターンを調査、検討した結果をとりまとめたものである。

汎用プログラムに関しては、情報処理振興事業協会等に関する法律により「特定プログラム」が定義づけられているので、この定義の線に沿って検討を行なった。すなわち、先ず、法に定める「特定プログラム」の内容を具体的に、委託開発、貸付の対象となるべきプログラムの範囲を明確にし、汎用化の手法を示した。

ついで、特定プログラムの委託開発に関する基本設計から発注、検収、メンテナンスならびに開発されたプログラムの貸付けまでの一連の業務を実施する場合の標準パターンを検討し、図1-1の流れ図に示す標準プロセスを得た。

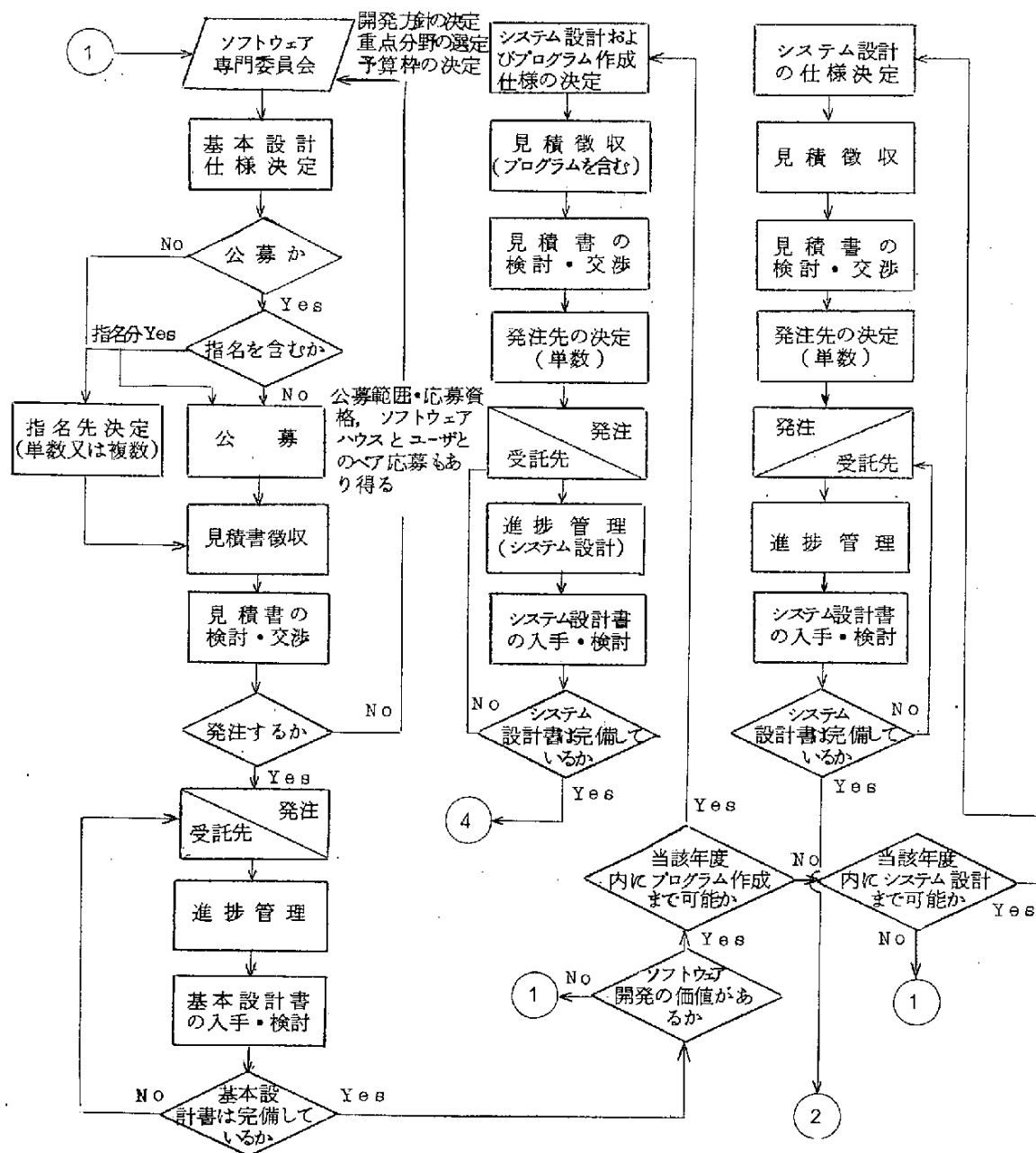
さらに、同図の各ブロックの内容、あるいはマイナ・ループで示されている行為の内容、たとえば委託開発のための事前調査、発注方法、委託発注価格決定の指針、委託開発業務の具体的な進捗管理方法、出来上ったプログラムの検査、検収ならびに貸付に伴う価格の決定その他の問題の取扱法、特にその際発生するプログラムのメンテナンス、あるいはプログラムの保証にまつわる諸問題を価格の問題も含めて検討し、標準的な取扱方法あるいは取扱方法あるいは取扱の指針を示した。

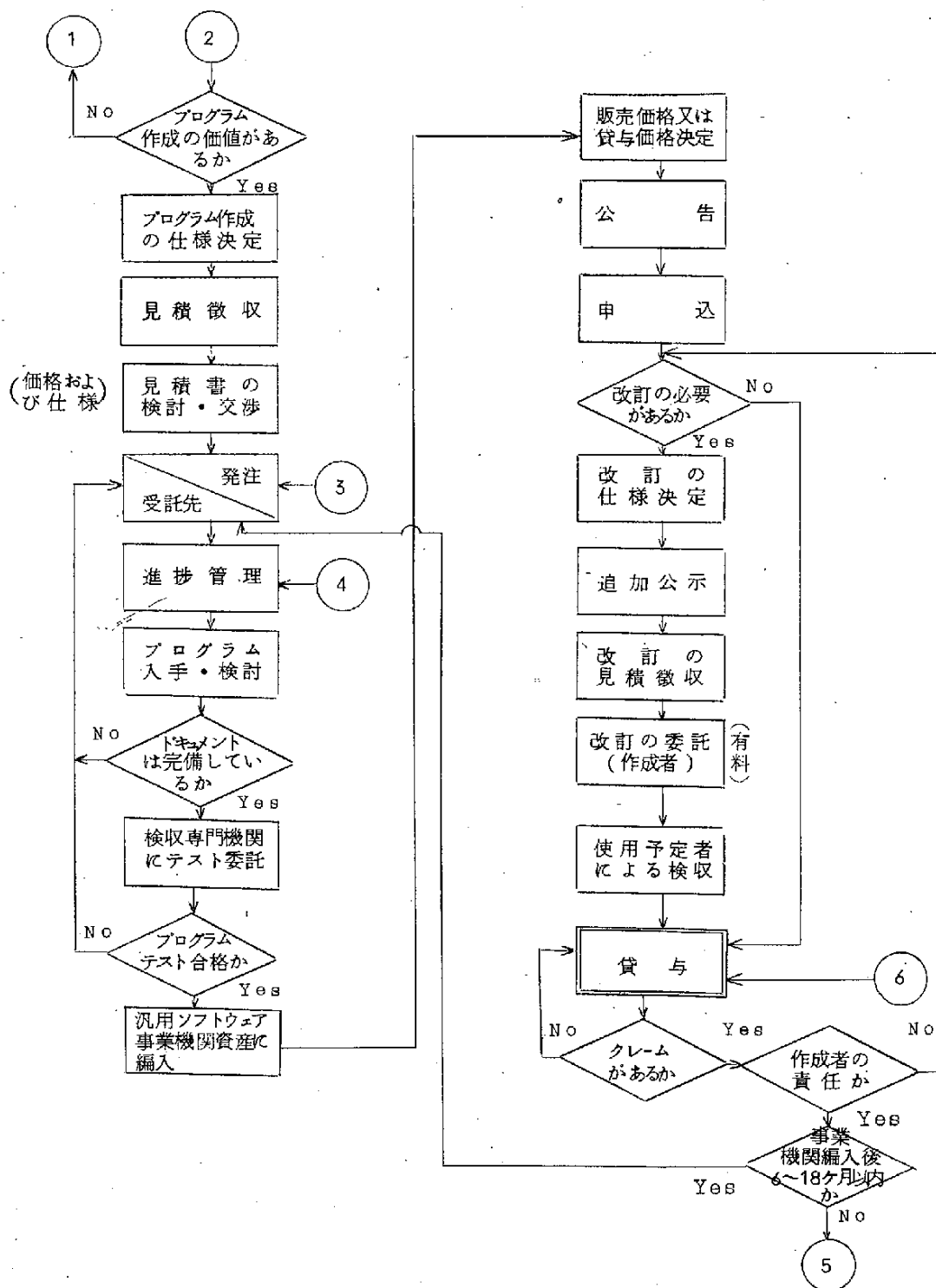
なお、今回の報告書に示したプロセスは標準パターンを示したものであり、業務の実施に当ってはこれを基準として適宜実情に合わせて修正する必要がある。

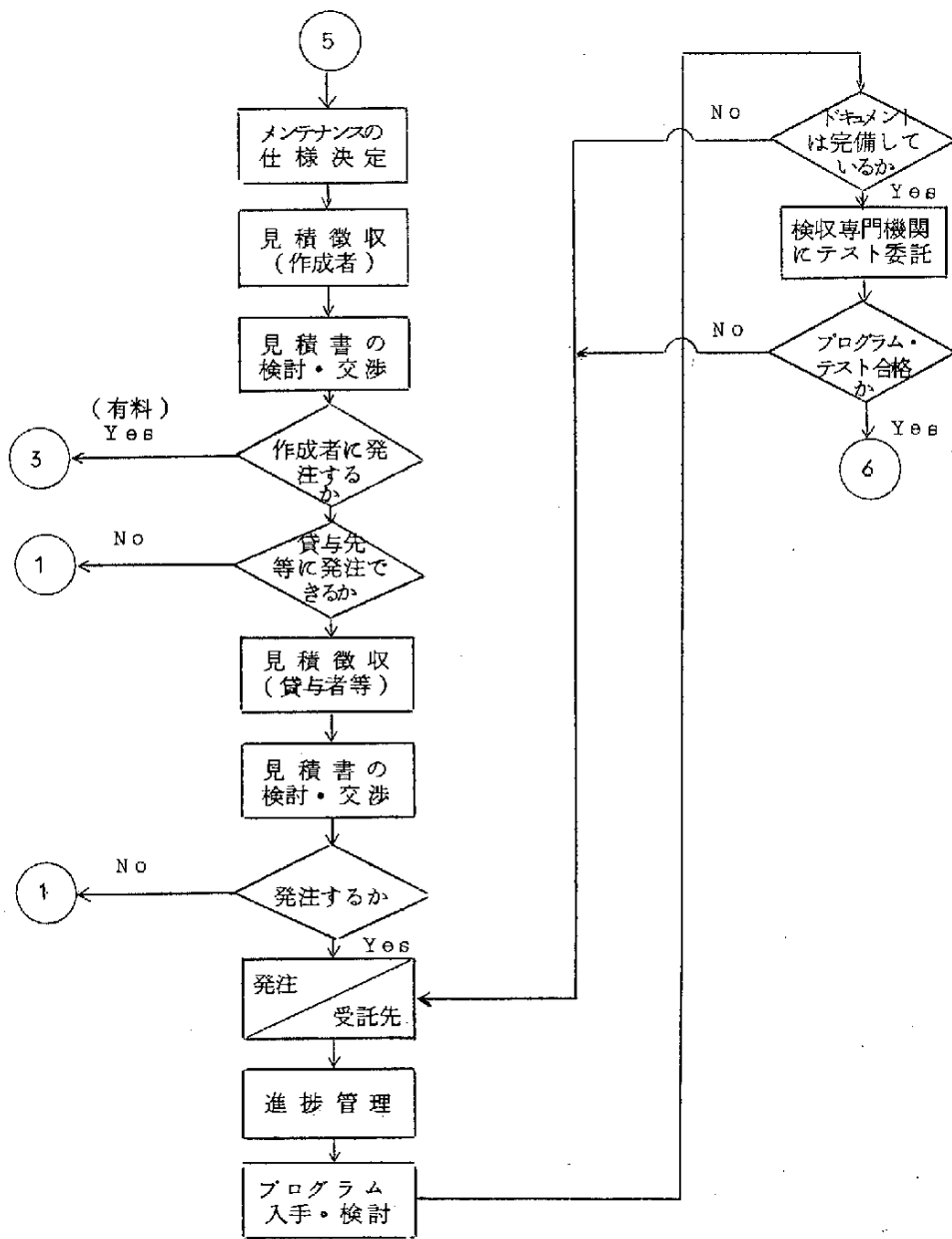
なお、報告書の内容は次の前提で審議したものであるから、この点、特に注意を喚起したい。

- (1) 図示の流れ図の出発点となっている事前調査の前に汎用プログラムの市場調査などの一般的な調査が当然必要であるが、これは周知のこととして報告書からは除外した。
- (2) この指針をとりまとめるに当っては汎用プログラムの委託開発、貸付等の業務を行なう汎用ソフトウェア事業機関に、プログラムの発注先の決定や出来上ったプログラムの評価、その他この種事業にともなう一切の業務を処理するソフトウェア専門委員会ともいべき組織があることを前提としている。

ソフトウェア発注から検収・貸与まで







2. 特定プログラムの類型化

2・1 プログラムの分類

2・1・1 目的から見たプログラムの分類

- (1) コンピュータ・システムをコントロールするプログラム（コントロール・プログラム）

今日のコンピュータは多数の機器や要素から構成された複雑なシステムであり、かつその用途もバッチ・モードからオンライン・リアルタイム・モード、TSSモードに至る種々の形態がある。従ってコンピュータ・システムの行なう仕事を用途に応じて円滑に遂行し、処理の能率を向上して応答時間を短縮し、ユーザにとって使い易いものにしなければならない。

このような役割を遂行するのがコントロール・プログラムであるが、このプログラムは以下のように構成されている。

- ① ハードウェア管理
- ② タスク管理
- ③ ジョブ管理
- ④ データ管理

- (2) 汎用プログラム言語をコンピュータ固有の機械語に翻訳するプログラム（言語プロセッサ）

コンピュータは固有の機械語によつてのみ仕事を遂行することができる。しかし人間が機械語によつてプログラムを作成することは極めて困難であり、かつ多大の時間を必要とする。この問題を解決するため種々

のプログラム言語が開発され、普及している。これらのプログラム言語には以下のものがある。

- ① アセンブリ言語
- ② コンパイラ言語
- ③ 問題向け言語

このうち、①はハードウェアと密接な関係をもつが、②③はハードウェアとは無関係である。これらの言語で書かれたプログラムをこれと等価な機械語のプロセッサに変換するために言語プロセッサが準備されねばならない。

(3) コンピュータ使用に際し普遍的に行なわれる仕事

コンピュータを使用するに際し、多数のユーザによって普遍的に実施される仕事が予めプログラムとして準備されていることはユーザにとって便利であることはいうまでもない。この種のプログラムはユーティリティ・プログラムまたはサービス・プログラムと呼ばれ、通常以下のものが含まれる。

- ① ソート／マージ
- ② リンケージ・エディタ
- ③ 各種ファイルの取扱い

(4) ユーザがコンピュータを使用してプロダクションを行なうためのプログラム

コンピュータを使用してユーザの行なう仕事は元来千差万別である。しかしながら取扱う問題が数学的に定式化され、または処理手順が論理的に厳密に設定され、比較的多数のユーザによって利用可能なものがある。これらをアプリケーション・プログラムと名付ける。これに反し特

定のユーザに固有のプログラムであつて他人が利用することの困難なものをユーザ・プログラムまたはカスタム・プログラムと呼ぶことにする。

2・1・2 プログラム作成者から見た分類

(1) コンピュータ・メーカーが作成し、ユーザに提供するもの

前項に述べたプログラムの目的から明かな如く

- ① コントロール・プログラム
- ② 言語プロセッサ
- ③ ユティリティ・プログラム

の3種類のプログラムはこれまでメーカーによつて準備され、ユーザに無償で提供されることが通例であつた。この内①は確かにメーカーが作成することを必須とするが、②③については必ずしも必須とするものではない。現にIBM社のように②および③については有償とする方針をとっている会社もあるし、ソフトウェア会社が作成してユーザに提供することもあり得る。

(2) コンピュータ・メーカーが営業政策上の見地から準備し顧客に提供するもの

多くのコンピュータ・メーカーがセールス・ツールとしていくつかのアプリケーション・プログラムを準備して顧客に提供することはすでに慣習となっている。たゞし、特定のソフトウェアは有償となっているものがあるし、IBM社はこの種のプログラムも今後すべてを有償とする方針を打ち出している。これらのプログラムには以下のものがある。

① サブルーチン型式のもの

科学計算用、統計計算用など

② 問題向け言語処理プログラム

シミュレーション用 (GPSS, SIMSCRIPT etc.)

数値制御用 (APT)

③ プログラム・システムとしてまとめられたもの

LPプログラム, PERT/CPM, 構造計算用プログラム,

情報検索など

④ ユーザのプログラミング作業を助けるもの

部品表展開

(3) ソフトウェア会社が作成し提供するもの

ソフトウェア会社が顧客の依頼により, または自主的に各種のプログラムを開発し, コンピュータ・メーカやユーザに提供することが米国においては盛んに行なわれている。

① メーカの注文により作成する特定のハードウェア・システムにマッチしたプログラム

言語プロセッサ, ユーティリティ・プログラム

② ユーザの注文により作成する特定の業務を遂行するためのプログラム

ユーザ・プログラムまたはカスタム・プログラム

③ 自らの責任において開発し不特定多数の顧客に提供するプログラム

ソート/マージ用プログラム

フローチャート作成用プログラム

情報検索用プログラム

会計計算用プログラム

(4) コンピュータ・ユーザが作成するもの

この種のプログラムはコンピュータ・ユーザの業種、対象業務により多種多様であり、かつ絶えず増加の一途を辿っている。

2・1・3 汎用性から見た分類

(1) 機種を問わないという意味でのプログラムの汎用性

一般に上位レベルの汎用言語で記述されたプログラムは機種を問わず汎用性があるといえる。ただし機種間によっては些少の修正を必要とすることがあり得る。もちろん対象とする計算機のメモリ（コア）の容量による制限があり得るし、データの表現形式の差による手直しを必要とすることもある。

またプログラムの性質によっては汎用言語と下位レベルの言語（アセンブリ言語）とを混用した例も少なくない。この場合は下位レベルの言語を使用した部分については機種に応じて書直しをすることが必要になる。

(2) どのユーザにも利用できるといった意味でのプログラムの汎用性

一般にユーザの使用する機種は多種にわたり、またコンピュータの適用業務は千差万別である。このような条件でユーザから見た汎用性のあるプログラムを定義することは容易でない。現在内外のメーカーが提供している主要な汎用アプリケーション・プログラムを参考資料－8に列挙する。

2・2 特定プログラムについて

2・2・1 特定プログラムの定義づけ

この報告書でいう特定プログラムとは国費等の公共費をもって開発するに値するプログラムをいい、以下の条件を備えるものでなければならない。

- ① 汎用性： プログラムに対する必要性が高く、広く利用活用され得るもの。
- ② 先進性： 高度の機能をもつプログラムで、その開発に多額の費用を必要とし企業ベースでは開発が困難なもの。
- ③ 採算性： 当初の需要は必ずしも明確でなく、採算性が不確定なもの。しかしながら3年程度後には需要が見込まれるものであることが望ましい。

2・2・2 開発すべき特定プログラムの選定方針

特定プログラムの開発方針を列挙すれば以下の通りである。

- ① コンピュータ・ユーザにとって効用の高いプログラムを優先する。
- ② 規模の大きいプログラムを積極的に開発する。
- ③ 中小規模のプログラムであっても利用頻度の特に高いものは開発の対象とする。
- ④ 汎用性の点から machine-independent なプログラムを優先する。
しかし部分的には machine-dependent になることはやむを得ない。

2・2・3 特定プログラム開発に関する示唆

開発の対象となる特定プログラムの選定に関し、目安となる事項をあげれば以下の通りである。

- ① サブルーチン型式のものについては、科学技術計算用の整備されたライブラリ・プログラム
- ② アプリケーション・プログラムでは規模が大きく、多額の資金を要するもの
- ③ ユーティリティ・プログラムとしては異機種間の変換用プログラム

- ④ 海外では開発されているが、日本では入手し得ないプログラム

3. 汎 用 化

特定プログラムは、なるべく多数のユーザに使用されることを目標としている。ここではプログラムにこの意味での汎用性を与える方法を考察する。現在世に出ているコンピュータすなわちノイマン式のコンピュータは、本来そのハードウェアが汎用性をもっており、これをソフトウェアによって個々の用途に適合させているので、ソフトウェアに汎用性をもたせることは、一般に難しい。したがって汎用化へのアプローチは、それぞれいくつかの条件をとらなっており、またそれらは相互に深い関係を持っている場合が多い。

以下汎用性のある特定のプログラムを得るためのアプローチと、これに関連する問題点をあげる。

3・1 汎用化へのアプローチ

3・1・1 一般的な利用目的に対するプログラムの汎用化

コンピュータを一般的な目的に利用する場合に、そのプログラム・ルーチンを、さらに広範囲の各種のコンピュータ・システムで利用できるようにするためには次の方法が考えられる。すなわち

- (a) プログラム記述用言語に汎用コンパイラ言語を使う。
- (b) すでに普及度の高いコンピュータのアセンブリ言語を使う。

3・1・2 特定分野向けプログラム用の高レベル言語の開発

ファイル処理用、一般事務処理用等、特定分野向けのプログラムを汎用化するため、これを記述する高レベルの言語を開発する。この言語に対する翻訳方式には目的プログラムが、

(a) 汎用コンパイラ言語たとえば、COBOLで書かれているもの

(b) 特定の機械語で書かれているもの

の2種類がある。

3・1・3 モジュール化方式の採用

(a) 設計仕様、またはフローチャートのレベルを標準化して、一般性をもたせる。(機能仕様書、またはフローチャートを商品化する)

(b) 通常のコンパイラ言語と、機械語の間に特別の中間レベルの言語を設定して、各種のプログラムの機械に対する依存性をなくす。

(c) プログラムをいくつかのモジュールに分け、そのなるべく多くを、汎用コンパイラ言語で書き、能率に影響する部分等を機械語で書いたモジュールとし、適用するシステムごとにこの部分を取りかえる。

3・1・4 ソフトウェアの汎用化を助ける技術の開発

プログラムそのものに汎用性を与えるアプローチ以外に、つぎにのべるように間接的にソフトウェアの汎用化、標準化を助けるプログラムを開発することも重要である。これらのソフトウェアは、作り方によっては汎用性のあるものともなり得る。

(a) 変換プログラム

普及度の高いコンピュータ・システム間で、それぞれのプログラムおよびデータに対する変換用プログラムを開発する。

(b) プログラムの開発、製作の自動化に使用するソフトウェア

フローチャート自動作成用、ドキュメント管理用、プログラムのテスト用等のソフトウェアを開発する。

3・2 標準化の推進

ソフトウェアの汎用化には、コンピュータのハードウェアの標準化が最も有効な手段であることは勿論であるが、その他の関連部門の標準化の推進も、極めて重要である。

すなわち、

- (a) コンパイラ言語の標準化
- (b) コンピュータ・システムにおける各種インターフェイスの標準化
- (c) 情報処理システムにおける処理形式、データ形式の標準化

なお、大規模なソフトウェア、能率を要求されるソフトウェアでは、コンピュータのハードウェアに対する依存性を無視することは、現実の問題として難かしい。これに対しては、3・1・3(c)でのベタモジュール化を最大限に活用することが重要であるが、他方製作の段階であらかじめ使用を予想される何機種かの異なるハードウェアを対象にして、それぞれの機種用のプログラムをセットとして、用意することが考えられる。

3・3 汎用性の評価

前述したように、特定プログラムの汎用性には、いくつかの条件が伴っている所以、その評価は難かしいが、ソフトウェアに対する一般的な評価の他に、つぎのような評価項目があげられる。

(a) 特定分野の範囲

特定の分野、または目的に対して設計されたプログラムでも、その範囲内でなるべく多くのユーザに利用されることが望ましい。したがって、汎用プログラムの持つ各種の機能がそれぞれユーザ側の事情による相違点を、どこまでカバーしているかは、汎用ソフトウェアの重

要な評価項目である。

ただし、これはプログラムの大きさ、ひいては能率との比較の問題で、考慮する必要がある。

(b) コンピュータ・システムとの適合性

入出力装置の種類、台数、性能に対して、選択の自由度の広いことが望ましい。

(c) オペレーティング・システムの規模に対する適合性

オペレーティング・システムでは、普通ハードウェア・メーカが修正を加えて、何度もバージョン・アップを行なうものである。したがって、汎用ソフトウェアは、オペレーティング・システムの中味に依存する部分が少ないことが望ましい。すなわち、オペレーティング・システムのバージョン・アップに対して影響を受ける部分がどの程度であるか、およびその部分の改訂（作り変え）が容易に行なえるか、ということが評価の対象になる。

(d) モジュール化の程度

プログラム・システムでモジュール化が正しく行なわれていることは、一般的に重要であるが、汎用ソフトウェアの場合では、そのバージョン・アップがやりやすいようにモジュール化してあることが必要である。

(e) ハードウェアに対する依存性

高いレベルの言語で書かれた汎用ソフトウェアは、原則としてその言語のコンパイラをもつハードウェア・システムでは使えるが、そのためには各種の条件が付帯している。それらの制約を考慮して、どれだけ多くの機種で利用できるかということが汎用性を評価する対象項目となる。

4. 委託のための事前調査分析

特定プログラムの発注について、その開発方針を決定し、発注価格を検討する際に必要な資料を得るために、事前調査が必要である。

機能の高い特定プログラムが開発され、これが広い分野で利用されるようになるためには、極めて広い範囲の多くの分野についての調査活動が必要であるが、ここでは、特定のテーマを決めてから発注に到るまでの間に、調査すべき問題およびその方法についてのべる。

なお、総論で述べたように、ここに記す事前調査にとりかかる前に予め汎用プログラムの市場調査等の一般的な予備調査が必要なことはいうまでもない。

4・1 事前調査の項目

発注するソフトウェアについて、事前に調査すべき最少限度の問題には、下記のようなものがある。

- (1) 汎用性
- (2) 市場の範囲および市場の要求
- (3) 利用される期間についての見通し
- (4) 競争品
- (5) 販売価格の見通し
- (6) 内容について
 - a. 設計の考え方
 - b. 適用についての範囲、制限事項

- c. 記述言語
- d. プログラムの大きさ
- e. ハードウェア・システム
- f. 原価の見通し
- g. 最少開発期間
- h. バージョン・アップ（更新）の見通し
- i. 関連技術

(7) 所有権の問題

4・2 事前調査の方法

4・2・1 事前調査の深さの段階

特定プログラムの内容によって、事前調査の深さは、次の各段階の適当なところまで行なう必要がある。その段階をプログラム作成に対して遠い所からならべると、次のようになる。

- a. 汎用性、市場性に関する背景（直接関連する一般的項目）についての調査
- b. 外部仕様設定、検収方法についての調査
- c. 発注条件等についての調査
- d. サブ・セット、または同種のものでの試行調査

4・3 事前調査の外注

事前調査は重要であるから費用をかけてでも実施しておかなければならない。事前調査を外注する場合には、そのテーマ、内容によって、下記のような方法が考えられる。

a. 特殊研究機関への依頼

b. 調査企業への発注

c. ソフトウェア会社等、汎用ソフトウェアの受注可能な企業へ発注

d. 汎用ソフトウェアそのものの納入企業に、その作業の一部としてやらせる。

どの場合にしても、調査項目についての詳細な仕様の設定が、調査の深さ、有効性に大きくひびいてくるので、これら調査についての要求事項を整理し、これに基づいて発注先から提案書（プロポーザル）をとって、十分検討した上契約を行なうことが望ましい。

4・4 事前調査と製作との関係

テーマの内容によっては、ソフトウェアそのものの設計、製作についてのプロポーザル、さらには仕様書の作成の段階までを含む場合も考えられるが、この場合には製作側の作業との重複を考慮しておく必要がある。

また、大規模なプロジェクトの場合には、事前調査と製作がある程度並行して行なわねばならない場合も生ずるが、この場合の製作作業に対する変更、または打ち切りを予想し、その対策を予めたてておく必要がある。

4・5 事前調査の費用の見通し

市場の動向、ソフトウェアの技術がともに急速に動いている現状では事前調査の必要性は高く、どの発注項目に対しても、ある程度の作業量（作業費）を見込んでおく必要がある。

また事前調査のみの委託発注も、取り上げられるべきである。

5. 発 注 方 法

5・1 発注の範囲

総論に述べたようにソフトウェア専門委員会は重点分野の選定、開発方針の決定および予算枠の決定など発注方針の基本を決める。

決められた発注方針に基づき、基本設計、システム設計およびプログラム作成の3つに区分し、それぞれを見積徴収したうえ、発注することを原則とする。（発注から検収までのフローチャート参照）

見積書は人件費、コンピュータ使用料、技術料及びその他の経費に分けて記述し、人件費については必要なマン・レベルとマン・アワーとを附記することが望ましい。

基本設計にはシステム設計の大筋までを含め原則として複数社に発注し、後続のシステム設計およびプログラム作成に関する発注作業の基礎資料とする。

システム設計及びプログラム作成については前記の基本設計書の内容を審査のうえ、原則として一社に絞り発注することが望ましい。

汎用ソフトウェア事業機関が会計年度制を採用する場合、当該年度にプログラム作成が完了する見込みのないものについては、システム設計とプログラム作成とを区分して発注するか、あるいは年度を異にするため一括発注分が途中で打切られることもありうる旨の事前諒承を得たうえ発注することが妥当であろう。

プログラム作成の委託発注については、プログラム利用上必要かくべからざるドキュメンテーションの作成仕様、プログラム検収に必要なテスト・

データ（最低５ケース）の用意，プログラム作成者の責に帰すべき事由による一定のサービス期間（６ヶ月～１８ヶ月）の取りきめなどを契約書中に明記しておくことが望ましい。

システム設計およびプログラム作成については契約時，契約６ヶ月後，検収後など適切な時期を定めて委託料の分割払条件を契約条項としておくことを要する。

５・２ 発注方法

発注は公募を原則とする。しかしながら，発注すべきソフトウェアは高度な頭脳の産物であり，開発期間が長期で，正確な見積り作業自体が困難であるうえに，プログラム作成が見積り通りに必ずできるとの保証も事前には予測できかねる場合もあるので，発注するプログラムのニーズの程度，内容，難易度によっては，応募者の提出資料，過去の経歴並びに人的，物的諸力の現状評価などを参考にソフトウェア専門委員会の議を経て指名発注することが必要である。

５・３ 発注方法上の問題点

(1) プログラムの委託発注の場合，下記のような事項をはっきり明示しておくことが必要である。

- ① 使用すべき言語およびそのレベル
- ② 利用するデータ・ファイルの構造と作り方並びに使用するファイル・ユニット名（磁気テープ，磁気ディスク）とその数
- ③ 期待する解の精度および範囲
- ④ 使用するコントロール・プログラム（OS，DOS，BPS等）の種類

- ⑤ 使用するコンピュータのユニット構成
- ⑥ エラー・メッセージの出し方およびその必要度
- ⑦ ビルト・イン・チェックの必要度
- ⑧ ダイアグノスティック (diagnostic) を添付する必要の有無
- ⑨ プログラム媒体 (磁気テープなど) の種類およびそのパリティ・
チェック方式などの指定
- ⑩ コメント・カード作成の程度
- ⑪ コンサルテーションおよびメンテナンスの無料サービス期間

基本設計に関する納入品は、システム設計およびプログラム作成を発注する基本的資料となるものであるから、プログラムという最終製品のあるべき姿を予想して、基本設計にはできるだけ多くの問題点を盛り込んだものを提出してもらうことが望ましい。

- (2) 応募資格や受注の条件として、あるいはまたメンテナンスの確保等のため、一定額の保証金を徴収することも考えられるが、書類審査その他の検討をきびしくすることで、これをカバーしうるものであるから必ずしも必要ではない。

6. 対 価 の 決 定

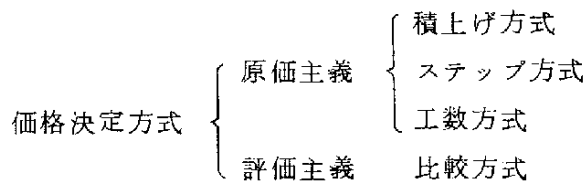
6・1 内外における対価決定の現状

ソフトウェア開発における対価決定の算定方法は未だ確立されていない。現段階では過去の経験に基づく勘と交渉 (Negotiation) により行なわれている。この状況は (財) 日本情報処理開発センターが昭和43年、44年に実施した「情報処理サービス業の実態」および日本電子工業振興協会が昭和42年に発表した「ソフトウェア価格に関する答申案」からもうかがえる。

ソフトウェアは頭脳の産物であるため、マンパワー・コストに吸収されるべき要素すなわち、開発すべきプログラムの種類、性質、技術、環境条件、処理能力、拡張性等は定量的に把握にくく評価は非常に困難である。また、保守、要員訓練等も考慮した場合、価格決定はより複雑になる。

米国においても、対価決定の多くがマンパワー・コストとコンピュータ・コストを中心とした直接費、間接費で算定されており、IBM, SDC, ディーボルト, ロッキードをはじめ最近では多くの会社が作業管理システムを確立して工数見積を行ない、それに工数単価を乗じて対価を算定している。しかし、価格決定となると未だ開発側と受入れ側との交渉によるところが大きい。最近では日本のメーカーやユーザでもこのような作業管理システムに力を入れつつある。

現在行なわれている価格決定の方法を体系化すれば次のようになる。



原価主義

開発側が作業を行なう上で発生する原価をいかに正しく回収するかという開発面に主体を置いた原価計算的な算定方法。

評価主義

周囲にある同種のプログラムと比較して有効性や利用価値を判断する方法。

積上げ方式

各作業プロセス毎に発生する経費を集計してゆく算定方式。

ステップ方式

過去の経験から1ステップ当りの所定単価を決定して、それに予測したステップ数を乗じて算出する方式。

工数方式

プログラムの量，難易度，計算機構成等の査定基準を設けて工程毎に各基準に対応するパラメータを選定して一定の方法で工数を算出する方式。

比較方式

ソフトウェアの内容や効果を過去に作成したものと比較検討したり，他社における同種のプログラムと対比して例えば，価格50万円と決定する方式。

6・2 対価決定に対する考え方

6・1 節内外における対価決定の現状でもふれたように、現在のところ決定的な対価算定の方式は確立されていない。従って如何なる価格で発注するかは発注方式によって異なることもあり、ケース・バイ・ケースで考慮しなくてはならない要素も多いと思われる。

そもそもソフトウェアの価格は、その利用価値により決定されるべきものであり、利用範囲の広いものについては巨大なソフトウェアで多額の資金を投入するものであろうと、少ない費用で開発されたものであろうと、内容の良いものには十分な対価を支払って買上げることが考慮されるべきである。

しかしながら、汎用ソフトウェア事業機関が発注するであろう特定プログラム開発プロジェクトは対象となるプログラムが未開発で、それと比較できるものは存在しないであろうから、価値決定の方式として特に、不確定要素の多いと考えられる評価主義をとることは適当でない。従って、それに代わる価格算定の方法として委託先から提出のあった見積りに対しては、その費用を十分に保証する原価主義を勧めたい。というのは、現段階では汎用ソフトウェア事業機関が行なう委託開発の目的には多分に情報処理産業の育成の要素が含まれており、発注を通してソフトウェア開発会社に力をつけることが必要なので、対価決定には十分開発費用を補填した原価主義をとることが望ましいと考えられるからである。

原価主義をとる主なるものには積上げ方式とステップ方式それに工数方式がある。ステップ方式は、単価の選定とかステップの予測がむずかしく不確定要素が多いため、経験や勘に頼っても、評価する人によりかなりの相違が生ずるのでむしろ好ましくない。また、工数方式は、各作業工程に

おける工数が正確に把握できれば価格の算定が簡便で実証性も高いので推奨したい方式ではあるが、まだ作業の標準化がなされておらず査定基準も確定的でない現状では信頼性に欠ける点がある。このように工数方式は、工程の標準化と査定基準が確立された段階では、もっとも望ましい算定方式といえるが、現状を考えると当面は積上げ方式を原則とすべきであろう。

積上げ方式は官庁等で受け入れ易い方式で、予算要求等にもこの方式が採用されている。また、発注方式でもふれたが特定プログラムの開発に国費等の公共費を充当する場合には、1年毎に区切りをつけて考えるべきである。プログラム開発が長期に及ぶ場合でも、発注は調査分析から保守まで委託先を一社に定めて一括発注することが望ましいが、上記のような場合には予算の性質上分割発注も考えなければならない。この場合にも分割自在な積上げ方式が便利である。このような観点からみても原価主義の採用が望まれる。

6・3 考慮すべき対価決定の要素

前節の6・1で原価主義による積上げ方式を推奨したので、この節ではその方式を中心に算定要素を考えてゆくことにする。

算定要素には横系列の経費項目と縦系列の作業工程とがある。さらに各々は次のように分類される。

経費項目

人件費、旅費交通費、資料費、会議費、消耗品費、計算機使用料、印刷費、一般管理費、外注費。

作業工程

システム設計（調査、システム設計）、プログラミング（プログラム設

計、プログラミング、データ作成、テスト・ラン、結果の検証）、仕様書・操作手引書の作成、保守。

プログラム開発の各工程では上記のような経費が発生する。その工程毎に発生する経費の見積合計額が開発に必要な対価である。対価は原則的に原価＋ α （利潤）で考え、利潤は技術料として評価されるべきである。

経費項目のうち直接原価的な旅費交通費、資料費、会議費、消耗品費、計算機使用料、印刷費等は予定数に単価を掛ければ求まるので比較的算出し易い。

ソフトウェアでは、そのものがもつ価値を技術料として認めるべきであるが、それを人件費の中で、個人のもつ技能をも含めた形のマンパワー・コストとして考えることができる。また、一般経費とは別枠の費用項目として認めることもできる。さらに技術料は調査研究費、教育費、その他等に分解して経費項目に組込むこともできる。外注加工費もこの考え方で既存の経費項目に分解して処理できる。

一方、別の見方として費用のうちのほとんどはマンパワー・コストとコンピュータ・コストであることから、これらの経費をまず算出し、その他の経費は一括して例えば全費用の20%というように決めてしまう方法もある。

作業工程においては、上記の工程をさらに細目に分割できるが、あまり作業段階を細分化（参考資料1-3）すると見積が面倒になるのでこの程度が適当と思われる。

保守の費用については、プログラム作成後の費用であるから別枠として考えてもよい。保証期間内で他の工程と同じ算定を行なってもよいが、過去の経験から保証料として全体費用の例えば10%と算出することも考え

られる。

6・4 対価決定方式の推奨パターン

6・4節では積上げ方式の算定方法および算定基礎について説明する。

対価算定の様式例は、表6-4-1に示す通りである。見積金額は、まず各工程で発生する経費を項目毎に集計し縦計をだす。その縦計に別枠で算定された保守料を加えて総額を算出する。

一般管理費は通信費、光熱費、不動産賃借料、調査研究費、減価償却費、雑費等の間接費であり、過去の通例から全費用の10%程度が望ましい。

人件費は給料諸手当、賞与、福利厚生費等で、その他に技術料を要員の技能手当としてここに含める。人件費1人1日当りの単価表例は表6-4-2に示す通りである。表6-4-2からは職種や経験年数から当然もつべき能力に注目して、仕様書や経歴書も参考に要員のレベルを決定する。

計算機使用料単価は $\frac{\text{レンタル料金/月} + \text{電力空調費等/月} + \text{関係諸経費/月}}{200\text{時間}}$
または $\frac{\text{レンタル料金/月}}{100\text{時間}}$ で計算する。

汎用ソフトウェア事業機関が開発するプログラムは先進的なプログラムであるから、その保証期間はできるだけ長期が望ましい。そのための保証料は対価の中に十分組入れておく必要がある。その意味からも保守料は過去の実例から総額の10%程度で計上したい。

各経費の算定基礎例は表6-4-3に示す通りである。

なお、その他の処理方式および算定基準は参考資料-1を参照されたい。

表 6-4-1

対 価 算 定 の 様 式 例

経 費 作業区分		人 件 費	旅 費・交 通 費	資 料 費	会 議 費	消 耗 品 費	計 算 機 使 用 料	印 刷 費	一 般 管 理 費	合 計
シ ス テ ム 設 計	調 査	○	○	○	○			○		○
	分 析	○	○	○	○	○				○
	シ ス テ ム 設 計	○			○	○				○
プ ロ グ ラ ミ ン グ	プ ロ グ ラ ム 設 計 (フ ロー - チ ャ ー ト 作 成)	○			○	○				○
	プ ロ グ ラ ミ ン グ (コ - デ ィ ン グ , デ ィ バ ッ ク)	○		○		○	○			○
	デ - タ 作 成	○				○	○			○
	テ ス ト ・ ラ ン	○				○	○			○
	結 果 の 検 証	○	○	○	○	○	○			○
仕様書, 操作手引書の作成		○			○	○		○		○
合 計		○	○	○	○	○	○	○	○	○
保 守										○
総 計										○

注) ○印は各作業区分で発生すると思われる経費

表 6-4-2

人 件 費 単 価 表 例

レベル	人件費/日	経験年数	技 能 水 準
1	17,500~35,000 円	7 年 以上	専門的な知識と技能を有し, 極めて高度な問題解決の援助と指導ができる者
2	14,000~17,500 円	5 ~ 7 年 位	与えられたプログラムのシステム分析およびシステム設計ができる者
3	10,500~14,000 円	3 ~ 5 年 位	プログラム・ドキュメントが与えられた場合, 定められた言語によりプログラムを作成できる者
4	7,000~10,500 円	1 ~ 3 年 位	プログラムの指示によりプログラムの作成又はコンピュータのオペレーションができる者

表 6 - 4 - 3

算 定 基 礎 例

経 費	内 容
人 件 費	○円／日×○人×○日
旅費・交通費	○円／回×○人×○回
資 料 費	○円／部×○種×○部
会 議 費	
会 場 費	○円×○回
茶 菓 子 代	○円×○人×○回
消 耗 品 費	
フロチャート用紙	○円／枚×○枚
コーディング用紙	○円／冊×○冊
パンチ・カード	○円／枚×○枚
紙テープ	○円／巻×○巻
MT	○円／本×○本
プリント用紙	○円／枚×○枚
リボン	○円／本×○本
計 算 機 使 用 料	
コンピュータ	○円／分×○分
カード穿孔	○円／桁×○枚×○桁／枚
紙テープ	○円／桁×○レコード×○桁／レコード
印 刷 費	○円／部×○種×○部
一 般 管 理 費	総額の10%程度
保 守 料	総額の10%程度
総 計	各経費の合計

6・5 対価決定の問題点

種々の観点から、原価主義による積上げ方式が好ましいと考えられるので、その考え方、算定要素、算定方式等について考えてきたが、この節で

は対価を決定する際の制約上の問題点、運用上の問題点等について考えてみる。

まず汎用ソフトウェア事業機関が発生するソフトウェアはかなり大きなものであるから、受注前に発生する調査費用もかなりまとまった金額になる。そのような経費を事前調査費として、プログラム作成費とは別枠で考えるかどうかという問題がある。

また、保守に要する費用は保証料という形でプログラム作成費の中を含めて考えるべきか、別枠で保証期間内の実費精算を行なうべきかという問題もある。

技術料をどこに含めるかも問題の一つであり、マンパワー・コストに入れるか、別枠で考えるか、全体経費の中に組入れるかなどが考えられる。それをどうするかによってマンパワー・コストも単なる通常の人件費か、技術料も含んだものになるのか、コンピュータ・コスト以外の直接、間接費全てを含んだものなのか、等かなり違ったものとなってくる。

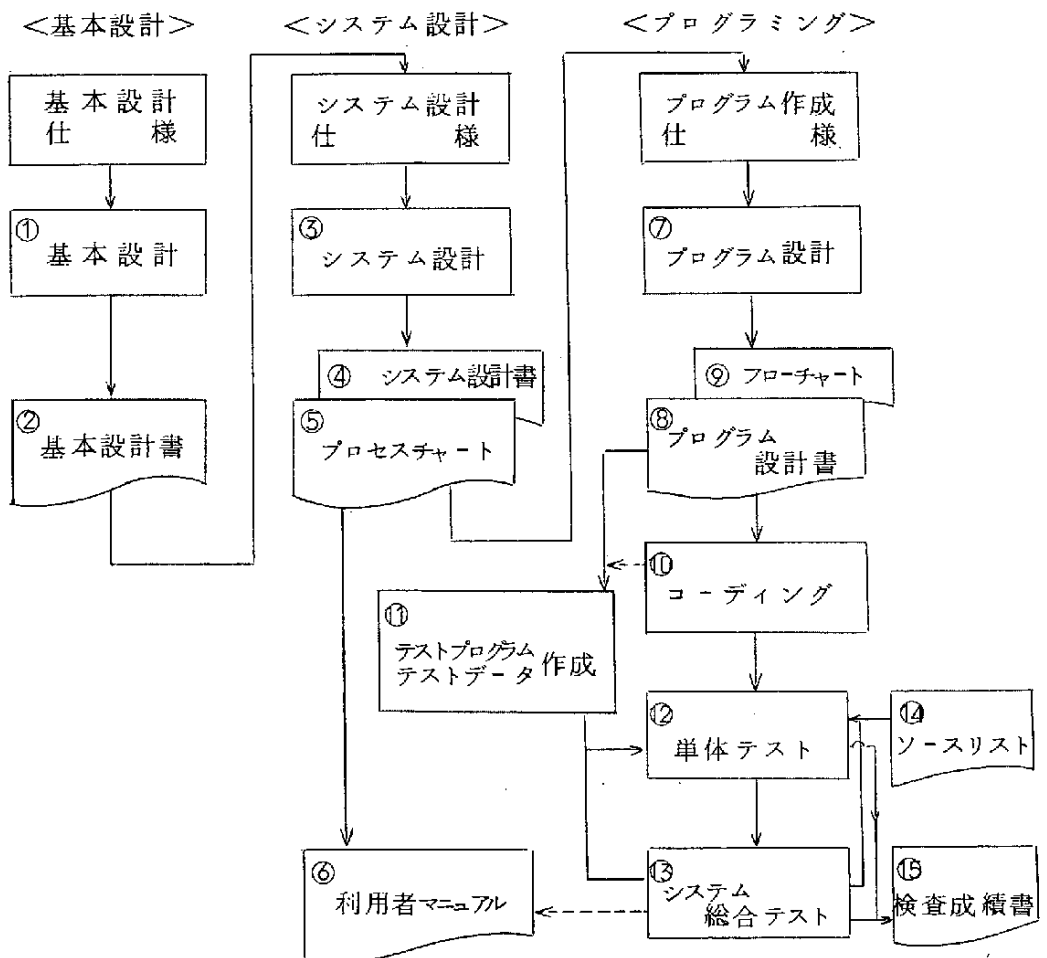
開発するソフトウェアはそのプログラムの種類、規模、技術水準によって Man Level, Man Hour, Computer Time 等の見積がむずかしい。過去の算定方式からみても各社各様であり、それぞれがかかえた制約条件によりいろいろな形で見積られている。このようなわけで算定方式の形態はつかめても代表的な例をあげ紹介することは難しい。なぜならば、そのケースがかなり特殊なケースであるかも知れず、それが代表的な例であるという確証は何もないからである。そこでこのようなものの評価、選定等にはかなり専門的な人達の協力が必要である。そのためにソフトウェア専門委員会の設置が望まれる。ソフトウェア専門委員会は、比較的プログラムに精通した人の集りで単純化した原価標準パターンをケース・バイ・ケ

- スであてはめ評価検討して対価決定を行なつてゆく権威ある委員会となるべきであろう。しかしこの場合の標準パターンもプログラムの生産性を示す参考尺度であつて絶対尺度ではない。この際には提出された仕様書、経歴書が十分に活用されなければならない。

7. 委託開発プログラムの進捗管理

7・1 進捗管理すべき範囲

ソフトウェア発注から検収、貸与までのプロセスは「総論」に於てフローチャートで示されているが、プログラムの設計、製作に関する部分を詳しく書き直してみると次のようになる。



プログラムの発注方式は既に5章に述べたように、原則として、基本設計、システム設計、プログラム作成の三つに区分される。

従って発注の際に提示する仕様には、区分を明確に、設計、作成のために必要かつ十分な事項を、洩れなくはつきり明示しておかなければならない。

進捗管理は仕様に記載された内容を満足する製品が、指定された納期までに納入されるよう作業の進捗状況を把握し管理することである。

それぞれの発注について進捗管理すべき範囲はつぎのとおりである。

基本設計については ①の作業および②の作成

システム設計については ③の作業および④～⑥の作成

プログラム作成については⑦～⑮の各作業並びにドキュメントの作成についてである。

は「製品」として納入させるべきドキュメントをあらわしている。

⑥の「利用者マニュアル」は、システム総合テストの結果改訂の必要が生ずることもあるので、この関係を点線で示した。

⑪のテストプログラム、テストデータ作成の作業については、システム設計書に記載されたデバック・テストに関する事項がすべてプログラム作成仕様に盛られることを前提としてここに位置づけた。

⑮の「検査報告書」には当然⑪の内容が含まれるわけであるが、一部を省略するようなことをしないで、作成および使用されたテストプログラム、テストデータを付属書類として納入させることが望ましい。

7・2 進捗管理の内容

ソフトウェア開発の進度を的確にとらえるための確実な方法は現在のと

ころ見つからず、設計、製作に関する管理方法は未だ一般化されていないようである。

ソフトウェア製作は、たとえ簡単なものであっても研究開発的な要素が多いから、本質的に一品生産となり、これが量産過程が存在しない一因をなしている。

ソフトウェアの開発には問題が多いが、その真の理由は、ソフトウェアに於ては「設計とは何か」という事が確立していない為であろう。開発の工程管理が出来るところまで「設計」という概念が固まることが必要であるが「設計」という概念の中には、出来あがるシステムの機能を如何に正確に、かつ能率よく実現するかということと同時に、それを如何に明確に記述するか、即ちシステム製作に関して如何なる種類の情報を有する書類を作成すべきかという二つの面がある。

ソフトウェアの「製品」は何かラベルを貼ったテープがあるのみで、製品の内容を知る唯一の手がかりは、いわゆる「ドキュメント」しか無い。設計書類なしにソフトウェアの内容を調べあげることは至難の業であり、不可能に近いであろう。

ソフトウェアの開発、特に売買という条件がつくならば、ソフトウェアの「製品」の定義は単にテープとかカードデッキだけでなく設計書類等ドキュメント一式を含めたものとしなければならない。即ち「製品」たるラベルを貼った数巻のテープ等は研究開発の成果の確認手段としては副次的なものと考え、システムに関してその内容ならびにノウハウを記述した一揃えの書類（システムの細部に亘る詳細な資料）が開発の成果であり、購入対象と考えられるわけである。

この考えに立てば、進捗管理すべき範囲、内容はおのずから作成納入さ

せるべき書類の種別，内容によって定まってくる。何故ならば，進捗管理にあたり直接確かめることが出来るのは各種ドキュメントと計算機からのアウトプットしか無いからである。

書類を種別ごとに内容について詳細に規定しておけば，全工数のかなりの部分がこれら書類の作成に費やされ，コーディングやデバックに要する時間は比較的僅少であり，かつ推計も割合容易で設計済み，書類作成済みのパーセンテージが全計画の進捗チェックの良い目安となり，管理もし易くなるであろう。

受注者に作成させるべき書類の種類には次のようなものがある。

(1) 基本設計書

(プロジェクト計画書，外部仕様書等)

(2) システム設計書

(構成および機能仕様書，システム仕様書等)

(3) プログラム設計書

(プログラム仕様書，内部仕様書等)

(4) ソースリスト

(5) 検査成績書

(6) 利用者マニュアル

これらの書類は記載すべき内容，書き方などを標準化し，誰にでも判るようなものにしておくことが大切で，それぞれの内容に応じた標準用紙をあらかじめ作っておくのも有効な方法である。なお，検査成績書の付属書類としてテストプログラム，テストデータも一緒に納品させるべきである。

7・3 工程管理の推奨パターン

既に述べたように、工程管理の標準パターンと称すべきものは未だ存在しないようである。

それに工程管理といっても、発注者としてのそれと、受託者としてのそれと二つの面がある。こゝでは発注者としての工程管理ということであるが、これも発注者がどの程度の規模陣容で、どの程度まで管理し、介入するかということによって、いろいろの形をとるであろう。しかし実際問題としては発注者が細部にわたり一々介入することは無理であろうから、マクロ的な管理にとどめ、細部については受託者に委ねる外はない。

この点からも、受託者の決定は提出諸資料、過去の実績、ならびに人的、物的諸能力などを厳正に評価し、信頼するに足る業者を選ぶようにしなければならない。

発注に際しては、作業スケジュール、製品の納期も基本条項として決められるわけであるから、これと実績との対比によって管理することになるが、具体的にはスケジュールの区切り毎に、あるいは一定期間毎にレポートを提出させ、それに伴う説明聴取によって判断する外はない。

詳細なスケジュールは受託者に作成提出させるが、これはPERT図を使用することが非常に有効である。これは実際に計算機で処理しなくても、図を書くだけでも有効である。即ち図を書くことによって、各プログラムや各作業の相互関係が明らかとなり、全工程に最も影響を与える作業が明らかとなるし、又作業を開始してから工程をチェックするチェックポイントも明らかになるからである。

このPERT図（フローダイアグラム）では各段階の作業を出来る限り明確に区分すると共に、その区分が達成されたことを明示する証拠物件が

何であるかを明確に指定しておくことが大切であり、又その区分が余り長大とならないよう、適宜チェックポイントを設けておくことも必要である。アローダイアグラムはどんなに入念に作成されても変更無しというわけにはゆかない。タイムリーに更新しなければ、その工程管理上の効率は非常に悪くなる。適切にバージョンアップやリビジョンアップを実施し、その経過の事由を明確にしておくことも大切である。

下の表は、製品である書類の納入時期の一例であるが、遅くともその時期までに納入させなければならない。

書類名 \ 作業段階	計 画	設 計	製 作	検 査
システム設計書	○	○		
プログラム設計書		○		
ソースリスト				○
検査成績書				○
利用者マニュアル			○	

納入手続としては、指定日までに作成完了した書類を逐次提出させ、必要に応じて説明を行なわせることである。

書類の納入後に於ても、設計に変更が生じたり、「結果を見て」のフィードバックが有効な場合もあるので、「納入済」のものに変更を加える必要が起り得る。このような場合には遅滞なく、先に納入した書類を更新させる。

従ってこれら書類のファイル形式はさし替え可能なものとするのが望ましい。

7・4 管理上の問題点、注意点

進捗管理がむづかしく、的確な方法が見つかりにくいのは、

- (1) システムの大きさを推定する規準の不明確さ
- (2) 作業区分の不明確さ
- (3) 作業量推定の不明確さ
- (4) システムのドキュメンテーションの時間推定の不明確さ

などのためであり、予定と実績に大きな差を生ずることも多い。進捗管理をうまく行なうためには何よりもまず

(1) 信頼できる業者を選定し、そこに発注することである。

- (2) 各エレメントに分けた詳細な作業工程表を提出させる。
- (3) それによって全体の仕事量を把握する。
- (4) 製品の納期を定める。
- (5) 検査体制を作り、文書化と標準化の徹底を図る。
- (6) 作業工程の区分ごとに、一定の方式で記載したレポートを提出して進捗状況を報告させ、必要に応じて説明させる。

長期にわたるプロジェクトの場合には、スケジュールの区切りでなくとも四半期ごと、あるいは隔月ぐらいの間隔で進捗状況を報告させる。

ということで管理するわけであるが、なお次の諸点に注意する。

- (7) モジュール思想を採用し、プログラム単位は大きくしないで小部分の組合わせとする。
- (8) 仕様書、設計書など、ドキュメントが未完成の作業については、発注を中止するか、あるいは作業の進行を停止させる。

(9) 方式決定にあたって、納期、性能、保守の面から検討して一方式をとる。この検討経過もドキュメントで提出させる。

(10) プロジェクトの進行につれ、機能、性能などの面から適宜システムを再検討すべきであるが、一旦決定したものの仕様変更は必ず発注者の承認を受けさせる。

なお完成予定日の一定期間前からは、絶対に仕様変更はしないようにする。

(11) 作業工程上異常を生じた場合には遅滞なく報告させ最善策を打ち合わせる。

(12) 要望事項とのくい違いが生じた場合には無償でかつ期限つきでやり直させる。

(13) 納期遅延の事態が発生しても、人員投入あるいは強化等の方法により遅延事態を回避出来るような分掌態勢をとらせておく。

などの諸点も進捗管理上大切なことである。

8. 検 査 ・ 検 収 方 法

8・1 検 査 ・ 検 収 の 範 囲

発注内容は、基本設計、システム設計およびプログラム作成に区分されるが、その各々に対して、納入品が発注仕様に合致したものであるか否かの検査をしたうえで検収手続を行なうものとする。

納入品がプログラムである場合は添付テスト・データに基き専門検収機関にテストを委託し、その結果の報告に基き受託者に検査結果の可否の通知、並びに修正を要する場合は修正依頼を速に行なうものとする。検査に合格した場合は汎用ソフトウェア事業機関資産に編入すると共に検収手続を行ない、代金支払手続を進めるものとする。

8・2 検 査 ・ 検 収 の 方 法

検査・検収の方法は、納入品の形式的・内容的書面審査による検査方法と、プログラム・テスト（テスト・データ）によるオペレーションを伴う検査方法とに分けられる。

形式的書面審査は発注仕様に合致するか否かを検査基準として利用する。

プログラム・テストによる方法は、専門検収機関をプログラムの内容・使用機器構成などによって適切に選択して、テスト・データ（通常5ケース）によるオペレーションの実施依頼を行ない、そのテスト報告に基き検収手続を進めるか、修正依頼を受託先にするかかの決定を行なうものとする。

8・3 メンテナンスの保証と期限

プログラムの使用上のニーズは予め想定できにくいので、メンテナンス範囲をなるべく狭く解釈する立場を容認のうえで、一応メンテナンス期間を6ヶ月～18ヶ月としておく。その場合当該プログラムがテスト・データによる検査に合格済みであれば、6ヶ月～18ヶ月以内は必ず無料で受託者が応ずるよう義務づけることは、プログラムという特殊な商品の性質ならびにソフトウェア産業の現状からみてなかなか困難であろう。そのため、図1-1のフローチャート中心に“作成者の責任か”というテストの機会を挿入しておいた。

従ってプログラムのバージョンアップ(大巾な機能の追加)はもとよりリビジョンアップ(小規模の機能の追加)についても指名発注などの便法を講じて、なるべく優れた汎用プログラムの作成を目標に、かなり弾力的な措置をとることが望ましい。

8・4 実施上の問題点

メンテナンスの保証期限の決定はソフトウェアの開発が個人の主観的判断や個人的創意を待つことが多く、まだ経験的定型化が行なわれていない。一般に、受託者側は保証期間、配員業務に拘束を受けるとか、要員の社外流出によって困難を生ずるかも知れないという理由で、長期のメンテナンス契約を好まない。一方、委託者側は受託の状況とは無関係に、再委託や再発注をできるだけ避けるため、なるべくメンテナンスの長期保証を望むであろう。

保証期限内でもメンテナンスを有料とする意見もある。その根拠は委託者以外のところにメンテナンス依頼をするとすれば、当然再発注と同じ程

度の額になると考えられる。したがって、メンテナンスは有料であっても、発注機関がこれを一括して取り扱った方が経済的にはマイナスを少なくするのであるから、情報処理振興事業協会等に関する法律の趣旨を生かして検収後の修正は、発注機関が責任をもって処理することの方が育成の実をあげることになる。そして若し、メンテナンスを有料とすることが困難な場合は、あらかじめメンテナンス料をプログラム作成費に組み込んで価格算定をすることなどの配慮が必要であろう。

ただし、ソフトウェアのメンテナンスというものがどういうものであるか第3者に示してもなかなか理解しにくいものなので、その辺のPRには十分努力する必要がある。

また検収の際、プログラムという商品の性質上、あらかじめ仕様に詳細を規定しにくいものであるから、納入品としてのプログラムについては、プログラム手法上、一般水準以下のところはないか、あるいは新しい工夫がなされているかなどの評価を併せ行ない、次の発注時の参考またはセールス・トークのポイントに利用することが望ましい。

9. 貸 付 け 方 法

9・1 貸付けの条件と手順

貸付けを行なう際に最も重要なことは、利用者に対してパッケージド・ソフトウェアの活用をし易くするための体制を整えることであろう。

そのためには次のような事項が要求されよう。

- (1) 広報活動の充実
- (2) 貸付け手続きの簡略化
- (3) 有用性高い多種多様なソフトウェアの保有
- (4) ユーザ・マニュアルの完備
- (5) 行届いた指導と保守サービス

ユーザがソフトウェアを要求する場合は、次のようなケースがある。

- (1) 概説書（ユーザ・マニュアル）のみ
- (2) 概説書，仕様書，操作説明書等のドキュメンテーション
- (3) ドキュメンテーションおよびプログラム

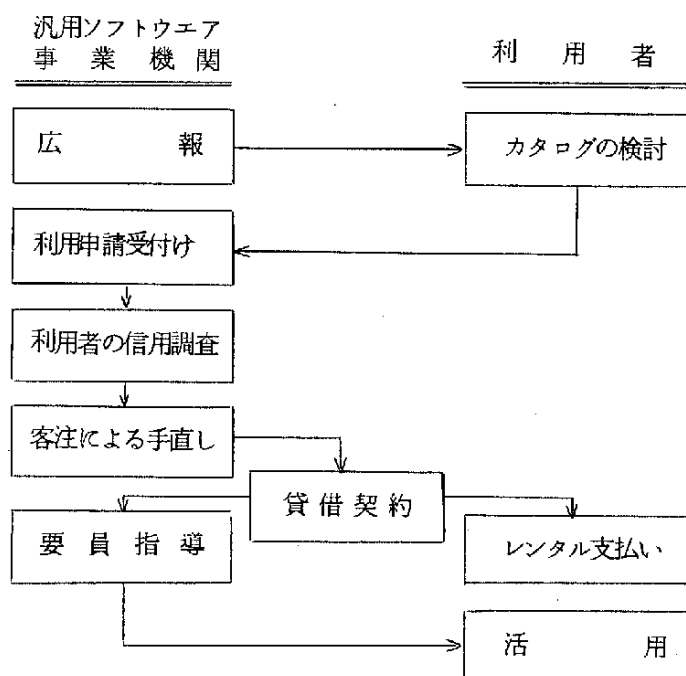
このうち貸付けが行なわれるのは(3)の場合であるが、ユーザが(1)，(2)を要求する場合はかなりある。このような要求に対して汎用ソフトウェア事業機関は、適当な対価でドキュメンテーションを提供することが必要であろう。

また汎用ソフトウェア事業機関は次のような態度で貸付けに臨むべきであろう。

- (1) 貸付けたソフトウェアは汎用ソフトウェア事業機関に帰属する。

- (2) 貸与するソフトウェアの転売，又貸しを許さない。
- (3) 許可なくプログラムの手直しをさせない。
- (4) 貸付け形式は原則として1年のレンタル制とする。但し期間を短縮するということも考えられる。
- (5) 2項の危険を防ぐため，利用者の信用調査を行なう。
- (6) 貸借契約をとりかわす。
- (7) 説明会，要員訓練等は使用方法を指導し，手法や基礎技術については教えない。その対象はある程度基礎知識をもつ専門家とする。
- (8) ソフトウェアに関する利害訴訟については，ソフトウェア専門委員会で検討のうえ適切な処置をとる。

ここでパッケージド・ソフトウェアが，利用者に活用されるまでのプロセスを示すと次のように画くことができよう。



9・2 貸付けの形式

貸付けの基本理念は、汎用ソフトウェア事業機関が保有する汎用ソフトウェアを出来るだけ多くの利用者に活用してもらうことにあるので、貸付けを円滑に行なうための形式を見出すことは重要な問題である。一般に貸付けの形式を大別すれば次の3通りが考えられる。

1. 汎用ソフトウェア事業機関が直接利用者に貸与する形式。
2. 特定の貸付け代行機関（ソフトウェア開発会社も含む）が利用者に貸与業務を行ない、汎用ソフトウェア事業機関に利用者を斡旋する形式。
3. 汎用ソフトウェア事業機関が特定の貸付け代行機関にソフトウェアを貸与し、貸付け代行機関が不特定多数の利用者に貸与する形式。

1のケース： この形式は汎用ソフトウェア事業機関が貸与に関するすべての業務を行なうものとして本来汎用ソフトウェア事業機関の指向すべき形ではある。しかし、新設の汎用ソフトウェア事業機関では貸与業務や保守サービス面で十分に体制を整えるにはかなりの期間を要すると思われるので、貸付けに関する一切の業務を単独で行なうことには困難が予想される。（図9-1）

2のケース： この形式では貸借契約を汎用ソフトウェア事業機関と利用者の間で結び、ソフトウェアは汎用ソフトウェア事業機関から利用者に提供される。貸与業務については代行機関にソフトウェアの内容説明だけを求め、汎用ソフトウェア事業機関と契約を直接することになるから、その契約が代行機関の活動によって実現したことを確認するため、汎用ソフトウェア事業機関が代行機関から常に貸与業務の報告を受け、その状況を把握すべきであり、汎用ソフトウェア事業機関は貸付代行機関に対し貸与

業務に協力した見返りとして、斡旋したソフトウェア規模や数に応じて手数料（例えば貸付高の１０％）を支払うべきであろう。（図９－２）

３のケース： この形式では、汎用ソフトウェア事業機関と貸付け代行機関の間で貸付けに関する権利の譲渡契約を結び、その契約の範囲内で、ソフトウェアの貸与および提供に関する権限を貸付け代行機関側に委せる。貸付け代行機関は規定のリベートを除いた貸付け料金を汎用ソフトウェア事業機関に納めなければならない。代行機関がソフトウェア開発会社の場合には、利用者が直接開発先と相談ができることになるので都合がよい。また汎用ソフトウェア事業機関にとっても貸付け代行機関との形式的な事務処理だけで済み、利用者との折衝が省けるので楽である。しかし、とすれば汎用ソフトウェア事業機関の監視の目がとどかなくなり汎用ソフトウェア事業機関、貸付け代行機関、利用者の間で利害関係も生じかねない。とくに注意すべきことには、９・１節でも述べた汎用ソフトウェア事業機関の姿勢としての転売、又貸しの禁止の条項にも反するので、この方式の採用には十分な考慮が必要であろう。（図９－３）

このようにいくつかの角度から貸付け形式を考えてみると、汎用ソフトウェア事業機関にとっては１のケースと２のケースを組合せた方式が最も効果的と思われる。（図９－４）

汎用ソフトウェア事業機関が自から利用者に貸与業務を行なうとともに、ソフトウェア開発会社を始めとする特定の貸付け代行機関に貸与業務を委せて汎用ソフトウェア事業機関に利用者を斡旋させる方式である。この方式では、あくまでも貸付けに関する活動の主体は汎用ソフトウェア事業機関にあり、利用者を斡旋する貸付け代行機関は認めても、決して転売や又貸しを許可することにはならないだろう。

図 9 - 1

第 1 の 形 式

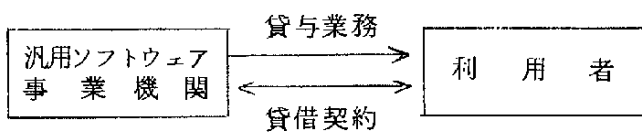


図 9 - 2

第 2 の 形 式

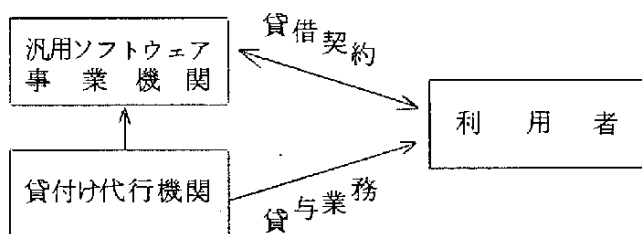


図 9 - 3

第 3 の 形 式

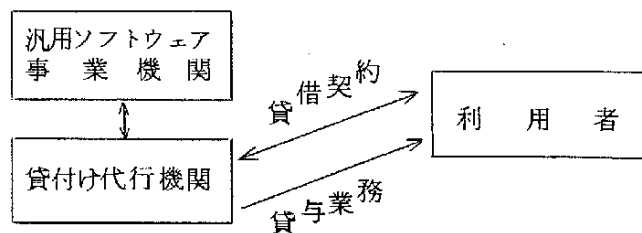
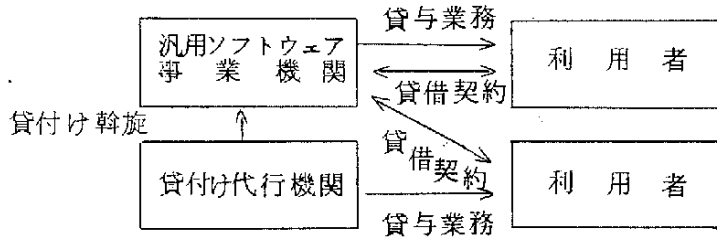


図 9 - 4

第 4 の 形 式
(第 1 と第 2 を併合した形式)



9・3 利用者に対するコンサルティング

コンサルティングについては2通りの考え方がある。その一つは汎用ソフトウェア事業機関が責任をもって当然行なわなければならない範囲であり、もう一つは利用者へのサービスとして汎用ソフトウェア事業機関が行なった方が好ましい範囲である。

前者では広報活動、導入指導、メンテナンス・コンサルテーションが主なものである。そのほとんどが無料サービスであるが、メンテナンス・コンサルテーションについては保証期間以後有料となるであろう。

広報活動は外部に対する活動としてカタログの作成配布、説明会の開催、電話による応答などが考えられ、内部活動としてユーザ・マニュアルの閲覧、利用に関する相談と指導等があげられる。

ユーザの理解を助けるためカタログやマニュアルには次のような内容が盛り込まれている必要がある。

- ① 使用目的
- ② パッケージの特徴と機能
- ③ 使用方法

- ④ パッケージの制約条件
- ⑤ 提供開始日
- ⑥ パッケージ導入に必要なステップと時間
- ⑦ これまでのテスト結果
- ⑧ 導入にあたっての費用とその後の運営費用
- ⑨ ハードウェアの構成例
- ⑩ ユーザが利用できるメンテナンス・サービスの種類

導入指導では、仕様書、操作手引書の説明手直しの相談等があり、メンテナンス・コンサルテーションは誤処理の指導、バグの補正等があげられる。

また、後者の汎用ソフトウェア事業機関が行なった方が好ましい範囲には、ユーザの注文による手直し、要員の教育、講習会の開催等がある。これらのほとんどは有料で、ケース・バイ・ケースに実費を徴収することになるであろう。

ユーザがパッケージド・プログラムを使用する際には多かれ少なかれ手直しが必要である。とくにユーザが使用するハードウェアからくる制約に伴う手直しは避けることができない。また、機能の改善、機能の強化等は必ず起る要求であり、汎用ソフトウェア事業機関はアフタケアの問題として、正当な対価を得てこれに応ずる必要がある。パッケージ使用に必要な要員の訓練と指導は、専門指導員の派遣により行なわれるであろう。

さらに、パッケージ導入を希望する利用者に対しては要求に応じて、オリエンテーションから詳細な内容説明まで、パッケージド・プログラム活用のための幅広い講習会の開催が必要となるであろう。

9・4 運用上の問題点

貸付けにおいて汎用ソフトウェア事業機関が最も配慮しなければならないことは次のような事項であろう。

- (1) パッケージド・ソフトウェアが予測数だけ見込通り貸与されているか。
- (2) パッケージド・ソフトウェア自体に欠陥はないか。
- (3) 広報活動やサービス活動が十分行なわれているか。
- (4) 用意したパッケージド・プログラムが利用者に即応できるものであるか。
- (5) 貸借契約上でトラブルは起らないか。

ここで、各事項ごとについて意見を述べる。

まず、(1)の事項； ソフトウェアの需要を見込むことは非常に難しい。パッケージの性質とか外部環境により、受け入れられ方によってかなり変わってくる。広報活動の充実もさることながら、貸付け回数の予測は比較的この方面に経験の深い専門家により編成されたソフトウェア専門委員会で検討されることが望ましい。

(2)の事項； ソフトウェアはその性質上、如何に慎重に検査検収してもバグの存在は避けられないものである。しかし、そのシステムがもつ機能を大きく損うような欠陥は、テスト・データを多く用いて取除いておかなければならない。

(3)の事項； そのソフトウェアが利用者に正しく理解されるか否かは、説明会やカタログの内容によるであろう。これらの内容は要領よく、理解し易く、充実したものでなければならない。また、バグや客注による手直しについては、速かにその要望に応じられるような体制を整えておく必要がある。

(4)の事項； 開発すべきソフトウェアについて，利用者の要求を反映できるものであるかどうかを，買上げまたは委託開発をする際にソフトウェア専門委員会で十分に検討しなければならない。

(5)の事項； 貸借契約は簡単明解なものが良いが，相互の利害関係を生ずるものの取扱については慎重を要する。転売や又貸しの禁止および違反に対する罰則，客注により手直ししたソフトウェアの所有権の問題等がその好例であろう。

10. 貸与するプログラムのメンテナンスの問題

10・1 プログラムのメンテナンス

新しく開発されたプログラムを、ユーザが使用する際にトラブルが発生することは避けられない。このような事態にも受託者側におけるプログラムのデバックおよびテスト、発注者側における受入検査に多くの時間と労力をかければ、トラブルの発生を減少させることは可能である。しかし、余りに多くの時間と労力をかけると、膨大な費用を必要とし、ひいては貸与価格が高くなるという結果を招く。他方、受託者側および発注者側では、予め予想し得ない種類のトラブルの発生もあり得る。これは結局のところユーザの使用実績に待つ以外に方法がない。

トラブル発生の原因をあげれば以下のケースがある。

- ① プログラム不良
- ② ユーザの取扱不良
- ③ プログラム設計仕様の不良または不備
- ④ システム設計仕様の不良または不備

10・2 貸与するプログラムのメンテナンス契約

前項に述べた理由により、プログラムが市場に出て、ユーザが使用している間は、プログラムのメンテナンスを必須とする。このため次の2通りの方法が考えられる。

- ① 汎用ソフトウェア事業機関側にメンテナンス要員を持つ
- ② 受託者と汎用ソフトウェア事業機関との間でメンテナンス契約を結ぶ

汎用ソフトウェア事業機関がメンテナンス要員をもつことは得策でない場合は②項の方式によらざるを得ないと考えられるが、その際には契約金額はプログラム開発の対価にリンクして決定することが当面妥当な策であろう。

10・3 貸与するプログラムの機能改善 (revision up)

プログラム仕様書は、実際にそのプログラムの実用時に起きるであろうすべての事態を予め洩れなく考慮することは不可能であり、実際の使用経験にもとづいて仕様を修正しプログラムの手直しをすることは往々起り得ることと考えられる。この場合、以下の手順で処理することが望ましい。

- ① ユーザからの要求などの情報をその都度受託者側に提出する。
- ② これらの情報を整理して、手直しのためのプロポーザルを提出させる。
- ③ プロポーザルを評価し妥当と判断されれば手直し作業を発注する。
- ④ 検 収

10・4 プログラムの機能強化 (version up)

一般にプログラムは幾多の使用経験が積重ねられ、その結果として大巾な機能の強化が要請される事態に立到ることは稀ではない。特に、よく使用されるプログラム程このような機会が多い。この場合、以下の手順が考えられる。

- ① 当該プログラムのユーザを組織し使用経験にもとづいて情報交換を行なう。
- ② プログラムの機能強化について意見を取りまとめ仕様書原案を作成

する。

③ 仕様書原案の審査検討を行なう。

④ 発 注

参 考 資 料

参考資料－１ 価格算定の諸方式

価格算定方式は大別して積上げ方式，ステップ方式，工数方式，比較方式等があるが，積上げ方式については 6.4 節で述べたので，その他の算定方式について説明する。

ステップ方式

ステップ方式は仕様書から業務内容を詳細に解析して，技術レベル，作成時間，プログラム・ステップ数を決定して，これに所定の単価を乗ずる方法である。

見積金額＝１ステップ当りの単価×ステップ数

このステップ方式のポイントは，ステップ数を予測する能力，プログラム作成単価を選定する能力を十分もっている技術者がこれら２つの要素を決定する必要があることである。

表１－１ プログラム作成単価（１ステップ当り料金）

言語 難易度		アセンブラ	コボル	フォートラン
１	かなり高度の技術を要するもの	200 ～350円	500 ～750円	500 ～2,000円
２	専門技術を要するもの	100 ～200円	300 ～500円	300 ～500円
３	一般的な技術で処理できるもの	50 ～100円	150 ～300円	150 ～300円

国内計算センター調査（JIPDEC）４４年

また１ステップの内容が，①調査・分析からドキュメンテーションの作成，保守までを含む料金であるのか，②一般に使われているシステム設計からプログラム引き渡しまでなのか，③コーディングとデバッグだけなのか，に

よってその単価もかなり違ってくる。②、③の場合でプログラム全体を見積るには積上げ方式と併用するとよい。

表1-1はプログラム作成単価表であるが、主要言語別に業務の難易度を考慮して1ステップ当りの料金を決定することになっている。なお単価はかなりの巾をもたせてあるが、作成の段階に応じて①の場合は上限近くを、②の場合は中頃を、③の場合は下限近くから選定してもらいたい。

また、中型電子計算機において、プログラマが1日当り作成できるステップ数等について電子工業振興協会「ソフトウェア価格に関する答申案」(昭和42年9月)を参考に表1-2から表1-4に掲載する。

表1-2 プログラムの種類

タイプ	プログラム
①	assembler, compiler, IOCS, control program
②	utility routine, subroutine
③	application package
④	on-line real time program
⑤	users program

〔注1〕 ここでいう中型電子計算機とは、金額にして4,000万円～2億5,000万円未満のものをいう。(日本電子計算機分類による)

〔注2〕 この場合のステップ数とは、ソースランゲージ (source language) のステップ数をいう。

〔注3〕 タイプ④の on-line real time program は、大規模な総合システムではなく、比較的小規模なシステムとする。

表 1-3 1日に1人が生産するプログラム・ステップ数

タイプ	ステップ数
①	5 ～ 25
②	10 ～ 40
③	10 ～ 30
④	10 ～ 20
⑤	20 ～ 60

〔注4〕 一般に、4K、8K等の小型機の場合は、1日当りのステップ数は中型機より多くなる。

中型電子計算機において、プログラマがプログラミングに要する時間と計算機使用時間 (machine debug 時間) との割合は表1-4の通りである。

〔注5〕 この場合の“プログラミング”とはシステムデザイン等は含まず、予めデザインされたシステムに従って、アイテムデザイン、コーディングのためのフローチャート作成からデバッキングまでをいう。

表 1-4 計算機使用時間のプログラミング時間に対する割合 (%)

タイプ	計算機使用時間の割合 (%)
①	3 ～ 8
②	4 ～ 10
③	3 ～ 8
④	3 ～ 8
⑤	3 ～ 8

〔注 6〕 一般に 4 K, 8 K 等の小型機の場合は, 中型機にくらべ, 計算機使用時間の割合は, 約 2 倍くらい多くなる。また, 大型機の場合は割合としては可成り少なくなる。

工 数 方 式

この方式はプログラム作成の各パターンにおける見積工数を計算し, その合計工数 (マンパワーとして人日で表わすことが多い) に工数単価を乗じて見積計算を行なう方法である。この方法で問題になるのは工数算出方法なので, そのいくつかのケースを紹介して行こう。

- ① プログラム作成のパターンをいくつか考え, 工程において各ケースがどれ位のマンパワーで処理できるかを計算する。

表 1-5 見 積 工 数 表 (人日)

プログラムパターン	プログラム設計			コーディング			デバッグ	
	アセンブラ	コボル	フォートラン	アセンブラ	コボル	フォートラン	アセンブラ	コボル
1	3	2	2	5	3	2	3	2
2	4	2	2	6	4	3	3	3
.....								
n	6	3	4	8	6	5	4	3
合 計	32	18	21	35	20	18	15	11

- ② IBM1401 (メモリー 8 K) テープシステムにおけるプログラミングにおけるプログラミングの作業標準
マクロ・ロジック, ミクロ・ロジック, コーディング, デスク・チェック
ング, テスト・アセンブリ, ドキュメント作成と引渡し各パターンにつ

きスケールを10段階、複雑さを5段階にわけ、各々のマトリックスにマンパワーをあてはめてゆき、さらに各パターンの要素毎のマンパワーを合計して総合表を作成している。工数見積りは各パターン毎に要素を選択して合計しても、総合表だけで行なってもよい。

IBM 1401のプログラム評価には3つのパラメータを用いている。第1のパラメータはコーディングのスケールであり、第2のパラメータはプログラムの複雑さであり、第3のパラメータは入出力装置の数である。そこで複雑さを客観的に評価できるよう

$$\text{複雑さ} = \frac{\text{スケールのファクタ}}{\text{入出力装置の数}}$$

で表すこともできる。A～Dのランクづけは統計の結果により決められる。

表 1-6 総 合 表

スケール 複雑さ	1	2	3	4	5	6	7	8	9	10
A	9.0	11.0	13.5	15.5	18.5	20.5	23.0	25.5	27.5	30.0
B	13.0	15.5	17.5	20.5	23.0	25.5	28.0	30.0	33.0	35.5
C	18.0	20.0	22.5	25.0	28.0	30.0	32.5	35.0	37.5	40.5
D	23.0	25.5	27.5	30.5	33.0	35.0	38.5	40.5	44.0	46.0
E	29.5	32.0	34.5	36.5	40.0	41.5	44.5	47.0	49.5	51.5

複雑さのファクタの度合は次の表のとおりである。

ファクタ	複雑さの度合
A	簡 単 (Simple)
B	適 当 (Moderate)

C	やや困難	(Medium Difficult)
D	非常に困難	(Quite Difficult)
E	極度に困難	(Extra Complex)
F	不可能	(Impossible)

③ Development Time Estimation (S D C Management Handbook
for the Estimation of Computer Programming Costs)

この見積りはプログラムの設計、コーディング、テストの工数（人月）を見積ったものである。

a) 10人月位の小さなプロジェクト

$$\text{所要工数} = 0.29X_1 - 0.47X_2 + 1.45X_3 + 1.13X_4 - 207X_5 + 475$$

b) 10人月～90人月のプロジェクト

$$\text{所要工数} = 7.22X_6 + 4.04X_7 + 2.64X_8 - 13.25X_9 - 9.27X_{11} + 66$$

ここで X_i は

- $X_1 = \begin{cases} 1 : & \text{使用言語が Compiler} \\ 0 : & \text{" Assembler} \end{cases}$
 $X_2 = \begin{cases} 1 : & \text{メモリ・サイズ 16K} \sim 100\text{K} \\ 0 : & \text{" 16K 以下} \end{cases}$
 $X_3 = \begin{cases} 1 : & \text{このプログラムと同種のプログラムを作った経験がある。} \\ 0 : & \text{" 経験がない。} \end{cases}$
 $X_4 = \begin{cases} 1 : & \text{タイミング・コントロールの要素がある。} \\ 0 : & \text{" 要素がない。} \end{cases}$
 $X_5 = \begin{cases} 1 : & \text{使用するコンピュータが初めてのものである。} \\ 0 : & \text{" の使用経験がある。} \end{cases}$
 $X_6 = \begin{cases} 1 : & \text{プログラム作成場所が 1ヶ所} \\ 0 : & \text{" 分散している。} \end{cases}$

$$X_{12} = \frac{\text{既に開発され使用できるサブルーチンのステップ数}}{\text{全体の推定プログラム・ステップ数}}$$

(X_{12} まで)

この方式によるサンプル実験では工数と計算機時間に関するコストはプログラムの大きさに関係し、他の要因には関係ないと見られていたが、実験結果によれば他の変数を加えることにより予測信頼度は23%増加している。これはステップ数よりむしろ、他の適当な予測変数を加えて使用する方がコスト見積精度を増加させることを示している。

効果的なコスト測定のためには情報を追加することが必要である。しかし、現状ではプログラムのデータ処理能力、複雑さ、確実さ、使い易さ、可変性というような測定し難い設計特性が多いため目標は達成しにくい。

④ 日本電子開発機によるプログラム見積基準

この基準はFORTRAN, COBOL等のコンパイラによる委託プログラムの価格決定に対する工数見積基準である。

表 1-7 ジョブ・ファクタ (新規プログラム作成の場合)

ファクタ 範囲	a 仕様の完備	b ドキュメント要求	c チェック・データ	d ロス・タイム
A	1.0 (コーディングとデバッグ)	0.0 (不 用)	0.0 (不 用)	0.0 (計算機 NEDで自前)
B	1.3 (ディテールフローからデバッグまで)	0.2 (I/Oのみ)	0.2 (大部分提供)	計算機 0.25 (NEDで) 外部使用
C	1.7 (ブロックチャートから仕様書作成まで)	0.4 (サブルーチン機能説明書)	0.3 (半分程度提供)	
D	2.0 (全工程)	0.8 (詳細説明)	0.4 (全て準備)	

見積工数 (W_2) = $W_1 \times S \times J \times F$

W_1 ; 1ステップ当りの標準工数で $W_1 = 0.25 \text{ H/ステップ}$ である

S ; 総ステップ数 $S = M_1 + M_2$

M_1 はステートメントカードの枚数でカード1枚を1ステップとする。

M_2 はコントロール・カードの枚数でカード1枚を0.6ステップとする

J ; ジョブ・ファクタ $J = a + b + c + d$

F ; ファシリティ (難易度)

表 1-8

難 易 度

難易度	プログラム・ステップ数 (S)
1.0	0 ~ 200
1.1	201 ~ 400
1.2	401 ~ 800
1.3	801 ~ 1600
1.4	1601 ~ 3200
1.5	3201 ~

この方式を利用した場合の過去における実績と見積の誤差は、10%以内であるという実験結果がでている。

比 較 方 式

従来から価格の決定は通常見積り料金、即ち過去の経験に基づく勘によりなされてきた。ソフトウェアは、計算機使用と頭脳労働からの産物であり、特にマンパワーがコストを左右する大きな要因となっている。このため業務の内容、専門技能、周囲の条件等により受注の段階で価格を見積ることは非常に困難である。委託を受けた側でも完成してみないと対価が算定できないのでは無責任であるから、実際に要する費用ではなく、特定の機械で標準的な能力をもつプログラマならこの位でできるだろうという漠然とした算定を行なっている。そのほとんどは周囲にあるプログラムや過去に作成したプログラムとの比較で価格が決定されている。

ソフトウェアがユーザに利用してもらえるかどうか、即ち市場性があるかないかはユーザの評価にあり、それが有用なので是非利用したいとなれば想像以上の高い価格をつけてでも買ってくれる。100万円の開発費をかけたものがユーザの人気を呼び、1件10万円で40件も50件も売れることがある。また、その半額の1件5万円という価をつけたために100件も150件も売れることがある。

ソフトウェアは1度開発したものに対してはそのコピーを提供するだけでよいので、少数の利用者で利益をあげるより、多くの利用者に安く提供して利益を上げる方が得策といえる。いわゆる薄利多売の精神である。また、あるプログラムを100万円で開発し200万円の値をつけて販売したが1件も売れなかった。そのために100万円、10万円、5万円と値を下げて販売したがやはり売れなかったという例もある。その理由はユーザがそのプログラムの価値を認めなかったからであり、活用できるかどうか不安であったからである。

このような意味では、ソフトウェアは水物であり人気があれば Autoflow のように 600 本も売れるケースがある。ソフトウェアの流通は、宣伝活動、制約条件、価格等によりかなり影響をうける。宣伝通りの効果があって、事実みんなが活用しており、さらに安ければユーザは買う気を起すのは当然であろう。このように比較方式は原価を無視した顧客本位の対価決定法である。即ち利用目的にあっていて安くて、使えるものとなれば誰れでもそのソフトウェアを買うはずである。アメリカで出回っている汎用パッケージの 67% は IBM のハードをベースにしているといわれているが、それらのプログラムのほとんどは 1,000 ドル以下という安い値段で売られている。

参考資料 - 2

一年間に一人が生産する プログラムの大きさと種類

-IBM USER 43年10月号より-

プログラムの種類	プログラムの例	プログラムの大きさ
単純なプログラム	簡単な応用プログラム	6,000~10,000 ステップ
基本が確立しているプログラム	Compiler SORT / MERGE	1,000~ 3,000 ステップ
未知な分野のプログラム	都市計画モデル 経営計画モデル	500~ 600 ステップ

(プログラム・エラーを除去し、完全なドキュメントを作る時間を含む)

参考資料－ 3

細分化された工程表

作 業 区 分	
シ ス テ ム 設 計	現システムの調査
	現システムの分析・評価
	新システムの設計
	システム仕様書の作成
	システム仕様書のトランスファ
プ ロ グ ラ ミ ン グ	プログラム方式の設計
	ファイルの設計
	プログラミング
	テスト・データの作成
	デバッキング（単体テスト）
	プログラム説明書の作成
	プログラム仕様書のトランスファ
テ ス ト お よ び 評 価	ファイルの作成
	システム・テスト（総合テスト）
	テスト・ラン（実データのテスト）
	システムの評価
サ ー ビ ス	オペレーション・マニュアルの作成
	オペレーションの説明
	ドキュメントの整備，編集及び印刷
	システム引渡し及びシステムの保守

参考資料 - 4

職種体系と要求される能力

要求される 能力 職種体系		企画及営業				技術管理力				デザイン能力				Set Up	教 育		知 識						
		全般的企画及管理	経 営 調 査	見 積 能 力	渉 外 能 力	新技術評価力	全般的技術管理	システム・デザイン 管理	プログラミング技術 管理	プロジェクト推進 管理	機械室運営管理	マクロデザイン	システム詳細設計	プロセス設計及 プログラミング	ドキュメント	セット・アップ	ドキュメンテーション	システム・エンジニア リング	システム・インス トール	プログラミン グ	全般的知識	販売するシステム 製品	専門技 能
マネージャ	ジェネラル・マネージャ プロジェクト・マネージャ	○	○	○	○	○		○	○	○		○						○		○			
エンジニア セールスマン	シニア・セールスマン チーフ・セールスマン		○	○	○	○				○		○						○		○	○	○	
エンジニア システム	シニア・システム チーフ・システム ジュニア・システム		○	○	○	○	○	△	○		○	○		○		○	○	○		○	○	○	○
プログラマ	シニア・プログラマ チーフ・プログラマ ジュニア・プログラマ			○	○	○	△	○	○		○	○	○					○	○	○	○	○	○
オペレータ	スケジューラ コンソール・オペレータ オペレータ									○								○			○	○	○
パンチャ	スーパーバイザ パンチャ									○												○	○
ドキュメン タリスト	ドキュメンタリスト アシスタント					○						△	△	○	○					○	○	○	○
庶務	チェッカ(窓口業務) 庶務																			△	○	○	○
リサーチ	シニア・リサーチ チーフ・リサーチ ジュニア・リサーチ					○											○			○			

注) ○必要とされる能力, △あった方が望ましい能力

ソフトウェア産業振興協議会資料

参考資料 - 5 パッケージド・ソフトウェアの販売状況

米国におけるソフトウェアへの投資額は、最近ではハードウェアへの投資額を凌いでおり、ますますソフトウェア・アプリケーションの要求が強まっている。ソフトウェア会社の商売は請け負い開発と専有権付きパッケージド・ソフトウェアの開発があるが、特に後者は70年代に賭ける商品として期待されている。

1968年のパッケージド・プログラムの売上げは2億7千万ドルとシステム・プログラムに対する投資額2億6千万ドルを上まわっている。それが1974年には25億6千万ドルにまでなることが予想される。

米国では最近、IBMがソフトウェアとハードウェアの価格分離に踏み切ったことも手伝ってソフトウェア会社を中心とするパッケージド・プログラムの開発および販売は一段と活発な動きを示している。

ソフトウェアといっても特別な販売方法があるわけではなく、結局、ソフトウェアの対価が如何に理由づけられて市場に出回るかということである。現状では詳細な経費の積上げを行なって対価を算定しても価格は売り手と買い手の交渉（話し合い）によって決められている状態である。しかしソフトウェア流通が、今以上に盛んに行なわれ、至るところで価格が決められれば、それら個々のケースが体系づけられて規準価格が決定されよう。関係業界でも最近では、正当な対価で販売されるところにソフトウェア市場の安定性と将来への発展が開けてくることを認識しはじめてきている。

パッケージド・ソフトウェアの流通は主にソフトウェア販売会社を通して行なわれている。ソフトウェア製作者とユーザを結ぶブローカとしてはICP社が有名である。同社はICP Quarterlyと呼ぶカタログ誌を発行して

ソフトウェアを紹介している。現在、米国のソフトウェア会社数は約1,000といわれているが、このうちかなりの会社がICPを利用している。カタログ誌には毎回1,500件以上のパッケージド・ソフトウェアが紹介されており2万～3万ぐらいの利用者がICPを何らかの形で利用しているといわれている。

ICP Quarterlyを分析すると、販売価格を提示しているもののほとんどが1,000ドル以下の安いもので275ドル～350ドルがとくに多い。これは日本の例であるが、価格分離に踏み切った日本IBMの有料プログラムは月間レンタル7,200円～105,000円と、これもまた比較的安い価格をつけている。

わが国における現状は（財）日本情報処理開発センターが昭和45年3月にとりまとめた「プログラム登録制度の現状と今後の課題」にもみられるように登録プログラム1510件のうちわずか13件しか価格がつけられていなかった。その理由は、ソフトウェア会社や計算センターのほとんどがこの登録に応じなかったことによるが、これらの会社がメーカの下受けや特定企業の注文による請け負い業務一本やりで、積極的にパッケージド・ソフトウェアを手掛る力に欠けていることを意味している。それはまた、米国にみられるような国家予算が背後に控えたビッグ・プロジェクトが少ないことや、電子計算機の絶対数が米国の1/11以下であり、汎用パッケージを受け入れるだけの需要が見込めないためであろう。

パッケージド・ソフトウェアの定義

汎用性を満足させることを念頭に設計されたコンピュータ・プログラムもしくはプログラムのユーザであって、不特定のユーザに答えるもの (Business Press International, The World of Packaged Software より)

このような状況からも、わが国のソフトウェア会社、計算センターに対して国としてはもっと力をつける必要があり、今春制定された「情報処理振興事業協会等に関する法律」による情報産業育成策の効果的運用に期待するところが大きい。

参考資料 - 6 価格決定に対する考え方

貸付け価格決定の基本的な考え方は、対価すなわち原価を回収することにある。汎用ソフトウェア事業機関の使命は国家的見地からソフトウェアの流通と情報処理産業の振興を図ることであり営利を目的としていない。

貸付け価格決定の焦点は、いくつの利用者が何回程度利用するか、利用期間はどの位かという利用者数、利用回数それにライフ・タイムの目安をつけることにある。

どれだけ貸与できるかが想定できれば、あとは開発に要した費用を貸付け回数で割り、1回当りの貸付け価格が決定できる。

延べ何回貸付けできるかの見込み数は、最も利用しそうな分野に的をしぼり確実なところを予測する。例えば開発費2,000万円のAというソフトウェアが、Bという分野を中心に活用され1年間利用10社、2年間利用5社、3年間利用10社と予測できたならば、

$$2,000 \text{ 万円} \div (1 \text{ 年} \times 10 + 2 \text{ 年} \times 5 + 3 \text{ 年} \times 10) = 40 \text{ 万円/年}$$

即ち貸付価格は1年1セット当たり40万円と決定される。

プログラムは利用者に利用し易くするため、貸付け価格は安価なものにする。いわゆる薄利多売の精神である。このような意味からも利用者数の予測は多めに行なった方がよい。ソフトウェアはその性質上、最初利用されなくても時がたつうちに値うちがでてくるということは、まず考えられないので貸付けは短期で考えた方がよい。貸付け期間は開発にとりかかってから5年以内、償却期間の目安を一応3年位におくべきである。

レンタルは原則として使用する計算機1セット当りの料金であり、複数で使用する場合はそれぞれのセット毎にレンタル料金を支払わなければならない

い。但し、同一会社、または子会社で同一プログラムを数セットに使用する場合は多少の割引きも考えられる。また長期レンタルの場合も割引きを考慮する必要があり、いずれの場合もケース・バイ・ケースで処理すべきである。

保証期間内の保守料は、価格の中に組み込まれているが、それ以後の保守については、料金を徴収する。また、コンサルティング、客注によるプログラムの手直し、要員教育、ドキュメンテーションの配布等についてはこれもケース・バイ・ケースで有料にて実施する。

参考資料 - 7

パッケージド・プログラムの価格例 - 1

ユーティリティ

(金額：万円)

プログラム名	用 途	買取り	(月) リース	提供 件数
AUTO FLOW	プログラム・ドキュメンテーション		13.6	600
TESTPAC	パッケージ・ソフトウェアテスト	397.8	18.5	75
AUDIT-THRU	レポート・ジェネレータ	32.4	10.8	30
SCERT	コンピュータ・システムの シミュレーション		51.0	53
DOCUMATIC	RPGドキュメンテーション			52
FORFLO	フォートラン フローチャーティング	17.3		33
XDUP	1130 ディスク操作	5.6		42
NSA	オブジェクトコードを アセンブラ言語に変換	11.7		22
CFY	デスク・コピーイング	1.8		24
PRINTZ & PRNT-1	最大速度で1132プリンタ をドライブ	7.2		16
INSTANT-COBOL	COBOLプログラムの作成	12.6		12
QUICK DRAW	ドキュメンテーション識別	205.2		130
SSTPAC	チャンネル・インターフェイスと周 辺機器用オンラインシステムの検査	27.0		8

パッケージド・プログラムの価格例 - 2

ライブラリ

(金額：万円)

プログラム名	用 途	買取り	(月) リース	提供 件数
	在庫管理 A4279	11		
	市場管理 A300.0	200		
	税 計 算 A400.0	300		
	需要分析予測 F30,000	210		
	分散分析 F385	100		

プログラム名	用 途	買取り	(月) リース	提供 件数
SIMSCRIPT	シミュレーション	540	18.6	40
QWICK QWERY	データの検索, 分析	972	32.4	20
MARK IV	ファイル・マネージメント	1260		190
SCORE	情報検索および報告	270		56
TELNETE	最適ネットワークシステムの決定	324		1
GPSS/360-NORDEN	シミュレーション	540		5
CONTRACT COST CONTROL	コストコントロール	1080		1
QUICK DATA	小企業向けデータ記録管理		21.2	2
FILE MASTER	大企業向けおよび科学記録管理		45	3

パッケージド・プログラムの価格例-3

アプリケーション・プログラム

(金額: 万円)

プログラム名	用 途	買取り	(月) リース	提供 件数
	人員予測管理 C800	8		
	生産計画の決定 C800	7		
	計画納期管理 A12,000	90		
	得意先元帳 A2,380	5		
	売掛金請求 C2,580	10		
	土木会社の トータルシステム A30,000	500		
財務情報報告システム	一般元帳会計	450	21.6	14
GENERALIZED PAYROLL SYSTEM	連邦政府, 民間会社で利用で きる給与計算プログラム	720		48
MONEY MINDER	勘定口座より収入, 支出をレ ポートする			12
ILA	保険証券マスタ・ファイル維 持および報告機能		28.8	8
CARTIFICATE OF DEPOSIT AND SAVINGS	預金, 貸付プログラムの検証	324		17

参考資料 - 8

主要な汎用アプリケーション・プログラム

最適化問題

Transportation Problem
Assembly Line Balancing
Travelling Salesman
Portfolio Selection
Process Optimization Program
Mathematical Programming System

シミュレーション

Dynamic Model
Simscrip
Capital Investment Program
General Purpose Simulation System
Continuous System Modeling Program

スケジューリング

CPM COST
CPM MAN
CPM TIME
PERT COST

統計関係

Component Analysis
Cross Tabulating
Factor Analysis
Tetracholic Correlation
Biomedical Computer Program

予 測 関 係

E P A Method

Census Method

Weighted Regression Analysis Program

Multiple Regression Analysis

Non-Linear Estimation Program

Forecasting by Econometric System

Linear Econometric Model Simulator

土 木 建 築 構 造

Influence Line of Continuous Beam

Influence Line of Continuous Truss

Influence Line of Frame Structure

Irregular Form Girder

Composite Beam

Arch Dam

General Rahmen

Response Analysis

Curved Girder

Frame Structure Analysis Program

機 械 工 学

Steam Table

Free Board

Flexibility Analysis of Piping System

Automatically Programmed Tools

電 気 ・ 電 力

Network Analysis Program

Electric Circuit Analysis Program

Block Diagram Compiler

化 学 工 業

Multi-Component Distillation Service

Chemical Engineering Information Processing System

そ の 他

List Processor

Large Matrix Inversion and Solution of Linear Equation

Simultaneous Linear Equation

Contour Map

禁無断転載

昭和45年9月発行

発行所 財団法人 日本情報処理開発センター

東京都港区芝公園21号地1番5

機械振興会館内

TEL (434) 8211 (代表)

印刷所 有限会社 タカミ・プリント

東京都港区芝1丁目2番15号

TEL (451) 6993番

45-Z 001

