

アメリカのプロジェクト・マネジメント

昭和48年3月



財団法人 日本経営情報開発協会

この報告書は昭和47年度における日本小型自動車振興会から
小型自動車競走法に基づく小型自動車等機械工業振興資金の
交付を受けて作成したものです。

序 章 米国における EDP の現状

1. コンピュータの設置台数と価格、性能の推移

インストール数	一台当りの平均購入価格	性 能
1951年 10台	\$ 3,000,000	\$ 25
1959年 3,000	\$ 1,200,000	\$ 2
1964年 16,000	\$ 700,000	\$ 0.23
1972年 85,000	\$ 420,000	\$ 0.016
1980年 300,000	\$?	?

(注) 性能は 10 万桁の計算コスト

コンピュータの利用台数は飛躍的に増大、80年代には300千台の予想、これに伴い1台当り購入価格は1972年には51年比 $\frac{1}{7}$ に低下、又計算能力コストでは1972年は51年比実に $\frac{1}{1,500}$ に低下している。

2. ソフトウェアのコスト推移

ハードウェア自体のコストは前表の通り著しく低下したものの、ユーザのソフトウェア開発コストは、この20年間で改善されずむしろコスト漸増の傾向にある。

	1マシンインストラクションを書く費用
1951年	?
1959	\$ 4.50
1964	\$ 5.30
1972	\$ 7.25

3. 利用者側のコスト構成推移

コンピュータの使用にかかわる利用者側のコストを大きくハードウェアのコスト、ソフトウェアのコスト、及オペレーション・コストの3つに分類して、その構成比率の推移をみると次表の通り、ハードウェア・コストは大巾に低下する反面それに代ってソフトウェア開発コストが急増の趨勢にある。

従って、今後の利用者側の課題は、いかにこのソフトウェアの開発コストを合理的なマネジメントによって節減するかにある。

	1953	1969	1971	1980(見込)
ハードウェア・コスト	80%	32%	25%	22%
ソフトウェア・コスト	8	30	35	43
オペレーション・コスト	12	38	40	35

4. ソフトウェア開発要員の動向（アナリストとプログラマ）

米国では初期の頃、システム開発の段階、即ち、問題分析→システム・デザイン→プログラミングの中で、システム・デザインとプログラミングは、同一人（プログラマ）が当るものと考えていた。しかし、今やこの考え方は誤りであり、別々な人～前者はアナリスト、後者がプログラマ～が当るのを良しとする考えが定説となった。

言い換えれば、システム・アナリストとプログラマは本質的に異なる適性があり、プログラマあがりのアナリストは適当でないとされている。

2) システム・アナリストとプログラマの適性比較

	アナリスト	プログラマ
学 歴	大学教育	ハイスクール
学 識	会計学、経済学 統計学	特になし
性 格	創造性 組織力 コミュニケーション・スキル	論理性 注意力 書く能力
進 取 の 性 気	外交的	内 攻 的

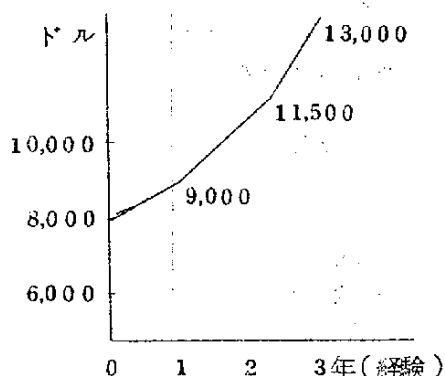
2) システム・アナリスト、プログラマの需給関係

米国のアナリスト、プログラマの需給バランスは完全に崩れており、要員養成ないし教育が急務となっている。

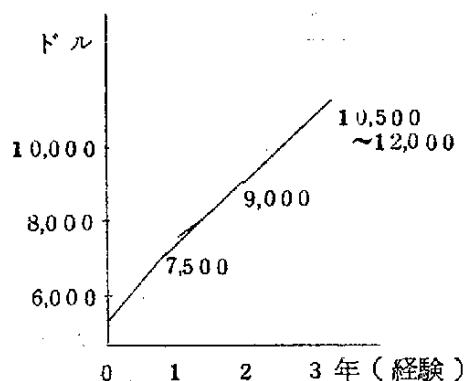
この結果、両要員の Salary Structure (給与体系) は他部門のそれよりハイレベルにあり、年々の昇給も高い。

	アナリスト	プログラマ
1969 需要	130,000	190,000
" 供給	90,000	140,000
不足	△ 40,000	△ 50,000
1970/71 需要	190,000	220,000
不足	△100,000	△ 80,000

アナリストのSalary(年間)



プログラムのSalary(年間)



- (注) ① アナリストの方がプログラマより給料は高い
 ② プログラムの初年度の昇給率は50%と高いのに注目

アナリストとプログラマの両者を比べると、将来はアナリストの不足の方が大きくなることが予想される。その理由はプログラマがコンピュータ言語の発展に従前の様な負担を感じなくなっていることとアナリストの方は下記いくつかの事由で逆にますます“分析時間”をとる様になって来たことによる。

- ① アプリケーションの範囲増大
- ② システムそのものの “ ”
- ③ デザイン技術の複雑性

アナリスト1人に対するプログラマの割合

	アナリスト	プログラマ
1958	1	10
1965	1	2
1972	1	1.1
1980	1	0.3

〔備考〕

コボル、PL/1 の出現
 バッケーン・プログラムの出現 (GIS・Mank IV・IMS)

現時点で1980年代のアナリストの不足数を予測すると、次の試算の通り約600千人に達する。なお、この数字は現在の米国の医者の数の約3倍にあたる。

1980年代のコンピュータ設置会社	175千~200千社	(1971、55千社)
1社当り平均アナリスト		千社 (" 実績)
1980年代の必要アナリスト	700千名~800千名	
1971年のアナリスト	10.0千名	
差 引 不 足	600~700千名	

5. EDP部門の組織パターン

EDP部門の組織には、会社の中に占める部門としての組織と、EDP部門の内部組織と二つがある。

1) 全社的な組織

次の三つのパターンがあるが、米国では(1)→(2)→(3)への進展がみられる。

(1)

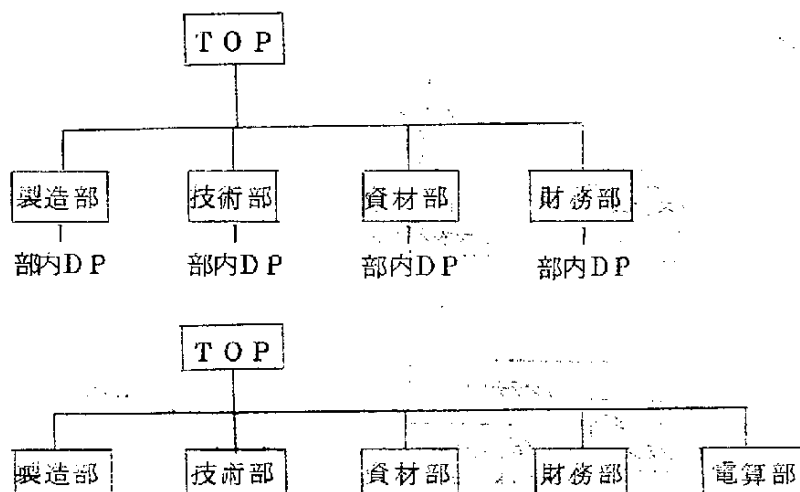
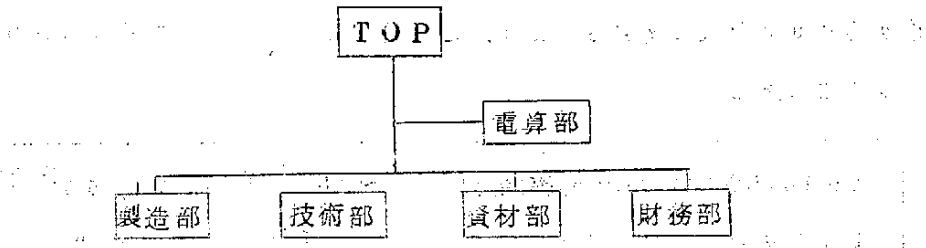
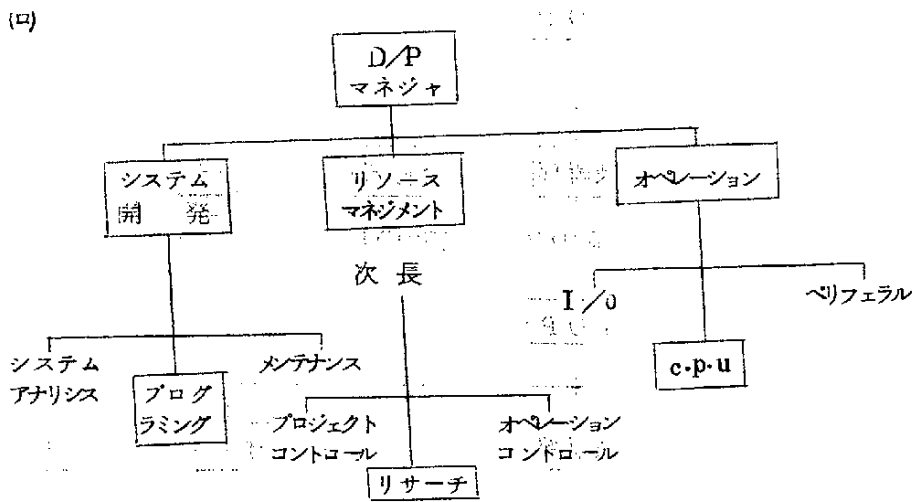
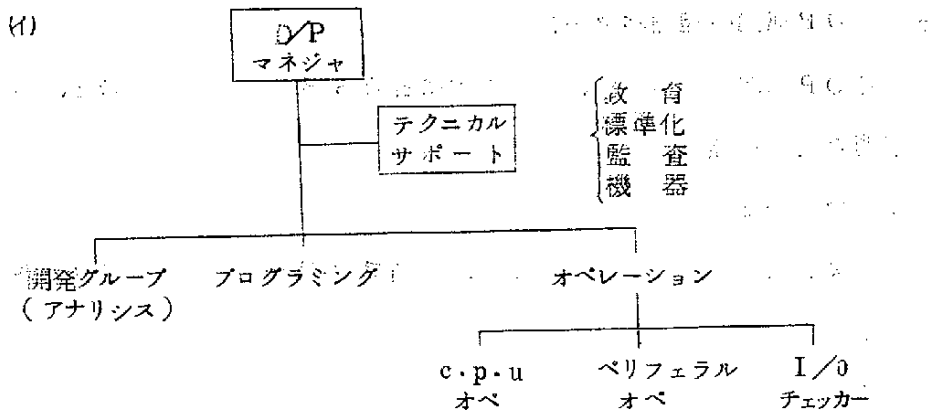


図 10



2) EDP部門そのものの組織

米国の現状は次図の(イ)の型が一般的だが、将来は(ロ)へ進むべきであろう。



上図の場合の各機能に注釈をつけておくと次の通りである。

D/P マネジャ

○ トップとの交渉
○ ユーザとのコミュニケーション

○ 長期計画の立案

装置については 5 ~ 7 年の計画を持つことが好ましい

人員については 3 ~ 5 年の計画を持つことが好ましい

○ 意思決定

リソース・マネジメントの機能

○ 人事管理

○ アナリスト・プログラマの教育

リサーチの機能

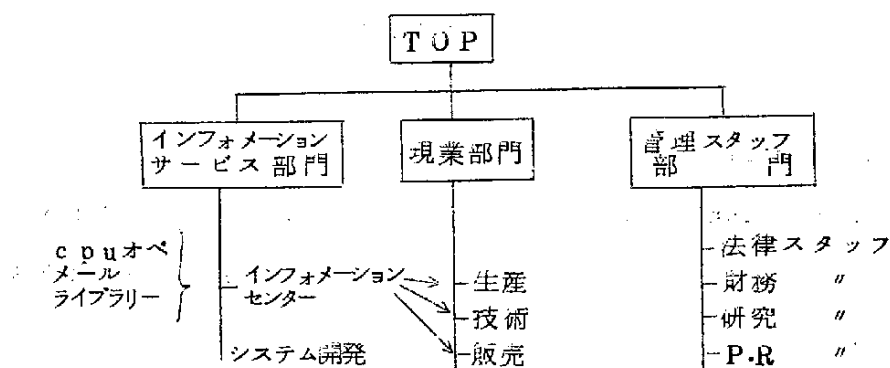
○ 装置の研究

○ ソフトウェアの研究

3) 1980 代の会社組織

E D P の発展、データベース・マネジメントの進展で、米国の会社組織は改革的な変化をみせていくだろう。

1980 代の組織を推定すると次のようになる。



第1章 インフォメーション・システムの序論

システム開発のステップに入る前に、システム全般の問題として、インフォメーション・システムのタイプとそれを処理するプロセスのタイプの二つについてふれておこう。

1. インフォメーション・システムの3つのタイプ

(1) Recordkeeping Systems

最も基礎的なシステムである。

特徴……1. Static files, Historicalな data

2. 意思決定要素はない

3. Batch control

定期的な更新、バッチ・インプット、定期的なレポートの作成

(2) Control Systems

企業内部の日々の業務遂行のコントロールと指揮とに積極的に参加するシステムである。

特徴……1. Dynamic files

2. 毎日の意思決定の資料となる

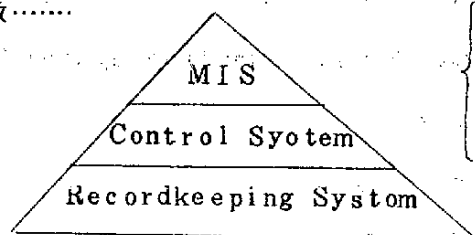
3. On-line control

(3) Management Information Systems

Information systemであってManagement systemではない。

長期的意思決定（経営の計画化、評価）をサポートする総合的なシステムである。

特徴.....



総合的なシステム
評価システム
外部データもインプットする。

(例) セールス・レポート
セールス・コントロール
マーケット・プランニン
グ

2. 処理形態

インフォメーションを処理する形態には3つのタイプがある。

(1) Batch Processing

大量データがグループ別に順序づけられ処理される

シーケンシャルまたはシリアルなプロセッシングである。

- Static files
- 定期的な更新、定期的なレポートイング

(2) On-line Systems

ダイナミカルに近いデータ・ファイルをもつ。

データ・ファイルのメンテナンスやアップデイトは他のプロセッシング・システムに依存する。

- Static files
- 自動的な Validation
- 毎日更新し、必要なときにレスポンスする。

(3) Real Time Systems

ダイナミックかつリアルタイムに、アクセス、更新、メンテナンスされるデータ・ファイルをベースとして開発される。

商業用のリアルタイム・システムは、しばしば Batch-Processing Operation に支援され、分離された On-line System を包含する。

- Dynamic files
- Direct Entry only
- 即時更新し、必要時にレスポンスする。

第2章 システム開発のプロセス

1. システム開発の13のステップ

一般に一つのシステムを完成するには13のステップを必要とする。

そのステップとは下表の通りである。

Phase	Product
(1) Application Selection	Problem Definition (Work Statement)
(2) System Survey	Design Requirements Statement, Systems Proposal
(3) Data Gathering	Definition of Existing System
(4) Data Analysis	Design Specification
(5) Data Base(File) Design	File Design
(6) Processes Design	Processes Design
(7) Systems Specification	Systems Specification Systems Specificationに対する documents
(8) Programming	Programs Programsに対する documents
(9) Systems Testing	Test Plan, Test Results
(10) Training	Training Plan, Training Aid
(11) Systems Conversion and Implementation	Conversion Plan Conversion Results
(12) Final Documentation	Maintenance Dossiers Historical Dossiers
(13) Post Implementation Evaluation	Reports Management — Project Evaluation — System Evaluation — System Maintenance

2. 各ステップの概要

ここでは各 Phase の概要について説明する。なお、具体的には Work Shop の中で説明を加える。

(1) アプリケーションの選択 (Application Selction)

(1) アプリケーションは企業におけるマネジメント・プランに基づき、組織のニーズを反映するさまざまな要因を考慮してセレクションされる。

—— EDP に関する長期計画をたてる。

—— 論理的にシステムの開発計画 (5 年間) をたてる。

—— 短期計画は毎年たて、たびたび変更がある。

(2) アプリケーションのセレクションもドキュメンテーションを行ない、組織の最高レベルの承認を得る必要がある。

—— ドキュメント

○ 組織的な計画表 (長期計画、短期計画)

○ ユーザよりのリクエスト (Work Statement)

—— 米国においては、アプリケーション・セレクションのための EDP 委員会を設け、この組織の検討結果に基づきトップが決定を下す。

(注) 何が重要であるかを協議し、優先順位を決める。

アプリケーション・セレクションを EDP 部門のマネジャに任せると、企業ポリシーと合致しない選択をしたり、プライオリティをつけ誤ることがある。

(3) EDP 委員会 (EDP Comittee または EDP Coodination)

—— EDP 委員会には、各部門からメンバーを出す。

—— 責任体制 上級マネジャ Senior Management

組織上のニーズ

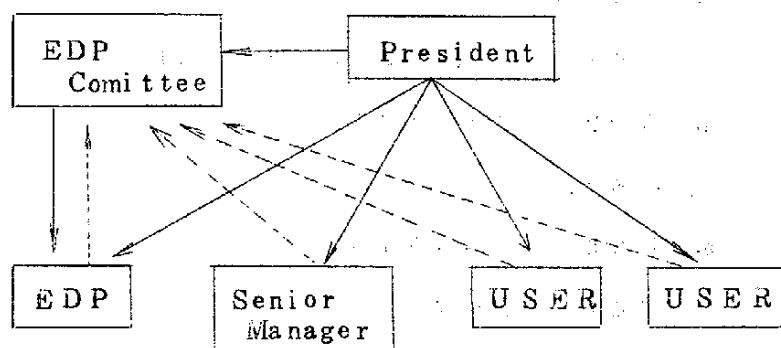
ユーザ部門 User Management

部門別のニーズ

EDP部門 EDP Management

取得可能なリソース・テクニカル・サポート

—— EDP委員会の組織



EDP部門は大体月に1度、EDP Plan(システム開発に
関するもの)などについて、EDP委員会へ上提する。

(f) セレクションのタスク

- a. 計画の作成
- b. ユーザのリクエスト集の作成
- c. 予備調査 → Work Statement

—— 問題の規定(明確化) } ユーザが行なう
 —— 問題の範囲を決定する }
 —— 時間、労力、リソースの見積り(予測)

—— 開発費の見積り（概算）

d. 仕事の割振り

e. EDP委員会かEDPマネジャがWork Statementを検討して、ユーザに必ず返事をする。

㈣ Work Statement

a. 要求者（仕事をやるもの）

b. 目的

c. システムの範囲

対象とするもの、しないものを明確にする

d. 開発の費用はどれだけになるか（概算）

コスト、時間

e. どのような便益があるか

特にユーザの利益

可能な利益（Potential Benefits）

—— EDP部門の責任として、開発部門、オペレーション部門のコストはできるだけ正確に明確にすること。

—— 投資に対する利益率はどうかについて、ユーザが決めるべきである。

○ EDP部門で予測することは不可能である

○ 間接的な利益ははかれないので人件費の節減のみを考える。

○ EDP部門における利益算出は、あくまでもその可能性であって、ユーザが決めるべきもの。

㍻ EDPマネジャは実際には、ユーザ側をサポートする立場に立つべ

きで、Application Selectionはユーザが行なうようにすべきである。

Application Selectionはシステム開発以前の問題ともいえる。

(2) システム・サーベイ (System Servey)

システム・サーベいは、開発プロセス全体の5～10%の労力を使う。

(1) サーベイの目的

- a. 詳細な問題 (要件も含む) の解析
- b. ちがったアプローチの方法 (代替案) の検討
- c. 改善の勧告
 - 最良の改善であるもの
 - 経済的なニーズを満たすもの
 - 技術的に可能なもの

(2) 生産物 (Products)

- a. デザイン要件記述書 — 「DRS」 (後述)
(Design Requirment Statement)
- b. システム設計提案書
(System Design Proposal)
—— ユーザ・マネジャ、シニア・マネジャの承認を受ける。

(3) サーベイ・タスク

- a. Work Statement を十分に理解する
- b. 予備調査の計画をたてる (Servey Plan)
- c. データ収集 (Data Collection)
- d. データ分析 (Data Analysis)
- e. デザイン要件記述書 (DRS) の作成
- f. 再検討 (ユーザの要件について)

g. 他の可能な方法（代替案）を定式化する（比較可能なもの）。

頭の中のデザインであり、最良と思うものを3つ程度にまとめる。

h. システム設計提案書(Proposal)の作成（必要ならば）。

i. 再検討（承認）（必要ならば）。

—— 以上のうちf、iはチェックポイントである。

(3) データ収集(Data gathering)（必要ならば）。

(4) データ分析(Data Analysis)（必要ならば）。

(5) データベースのデザイン(Data Base Design)（必要ならば）。

(i) システム・デザインの前提条件（必要ならば）。

a. デザインを早く始めると、変更するために変更するということになるので、データの分析が終了してから、設計を行なうこと。

b. 人員と仕事量のバランスを保たねばならない。

c. 積極的価値のある明確かつ簡潔な情報であり、実務上役に立つ情報であること。

d. システムに十分なコントロールを組み込んでおくこと。

e. 必要性と利益と費用のバランスを考えること。

f. システムに弾力性をもたせ、モディフィケーション、メインテナンスに耐え得るようデザインすること。

(ii) デザイン・タスク（必要ならば）。

a. システムの要件についての検討。

b. システム・フローの明確化。

c. システムの分割化

Sub-System-Segment化

d. ファイルのデザイン

- e. プロセスのデザイン
- f. スペシフィケーション
- h) システム・フロー (System flow)

- a. アウトプットの要件を明確にする

—— アウトプットのドキュメント化

- b. インプットの要件を明確にする。

—— インプットのドキュメント化

- c. データの項目 (Define Data Element) の明確化

—— Character

—— Data Element

—— Record

—— File

——→ データ項目のドキュメント化

- d. データ・フロー (Data Flow)

インプットからアウトプットまでのデータの流れを再確認する。

- e. システム・フローチャート (System Flow Charts)

〔例〕 NCR開発 ADS

(Accurately Defined System)

NCRで開発、NCR自身、Insurance Company of North America で適用、5種類のドキュメントを使用し、総括ドキュメントと残り4種類のドキュメントを相互関連させて、目的物のおかれた位置を明確にするドキュメント方法である。

翻訳NCR「ADSによるシステム設計」参照。

(二) アウトプット・スペシフィケーション (出力の仕様)

a. レポートそのものについての記述 (レポートそのものについて)

Description (記述) (目的、定義、特性、使用ひん度、使用ボリューム、配布先、複写枚数)

- 目的 (目的) (目的)
- ディフィニション (定義) (定義)
- 特性 (特性) (使用ひん度、使用ボリューム、配布先、複写枚数)
- 使用ひん度 (使用ひん度) (使用ひん度)
- 使用ボリューム (使用ボリューム) (使用ボリューム)
- 配布先 (配布先) (配布先)
- 複写枚数 (複写枚数) (複写枚数)

b. レポートのフォーマット、サンプル (レポートのフォーマット、サンプル)

- Output Format Chart (出力フォーマットチャート)
- Print Format (印刷フォーマット)
- サンプル (サンプル)
- Printer Control (プリンタコントロール)

(三) インプット・スペシフィケーション (入力仕様)

a. インプットそのものについての記述 (入力そのものについて)

- Identification (識別) (識別)
- 誰が作成するのか (誰が作成するのか) (誰が作成するのか)

- 作成ひん度、インプットのひん度 (作成ひん度、インプットのひん度)
- ボリューム (ボリューム) (ボリューム)
- インプットの媒体 (インプットの媒体)
- インプットの保存 (インプットの保存)

b. インプットのフォーマット、サンプル (インプットのフォーマット、サンプル)

- Input Format (入力フォーマット) (入力フォーマット)

1. ファイル・設計の考慮事項

(1) ファイル・デザインのタスク

a. ロジカル・エレメントのレコード化

b. レコードのグループ化

—— ブロック長、レコード長の考慮

—— 固定長か、可変長か

—— コンビネーション

—— プロセスのスピード

—— メインテナンスの難易度

—— アクセスの難易度

—— コントロール

長さの決定に考慮を必要とする。

c. ファイルの組立て、配列の決定

Up-date

Reporting

の関係をよくみて組立て、配列を考えるべきである。

d. ファイルの媒体の決定

—— Tape, Dram, Disk等

—— シーケンシャル、ランダム

—— ファイルのタイプ

Criteria ○ プロセスの方法

○ メインテナンスの難易度

○ コントロールの難易度

○ アクセスの方法

Factors ○ 更新の頻度

○ マスター・ファイルのレポートイング

ん度およびボリューム、データのボリューム

- 各更新時における Data Transaction とマスターとの一致する割合。

hit rateが低い(15%以下)ときはディスクを、高いとき(15%以上)はテープを使う。

e. Organize File

f. ファイル・スペシフィケーション

—— ファイルそのものについての記述

- Organization/Structure

- ブロッキング・ファクター

Label レコード

Header レコード

Trailer レコード

- ファイル・サイズの規模

- バックアップ(保存)

- 保全→アクセスしてよいもの

- プログラムはどのように使用しているか

—— ファイルのレイアウト

(6) プロセス・デザイン

(1) 機能

プロセス・デザインには2つの主な機能がある。

- a. 適当なオペレーティング・サイクルに分割する

- b. サブシステムに分割する (入力・出力・処理・データベース)

(四) プロセス・デザインによって作られる文書

- a. フローチャート

- b. プログラム仕様書

(五) プロセス・デザインの内容

- a. インプット・データ収集

- b. 編集と検証

- c. 更新

- d. 分類

- e. ファイル抽出

- f. 報告書作成

(六) サブシステムの種類

- a. インプット・サブシステム

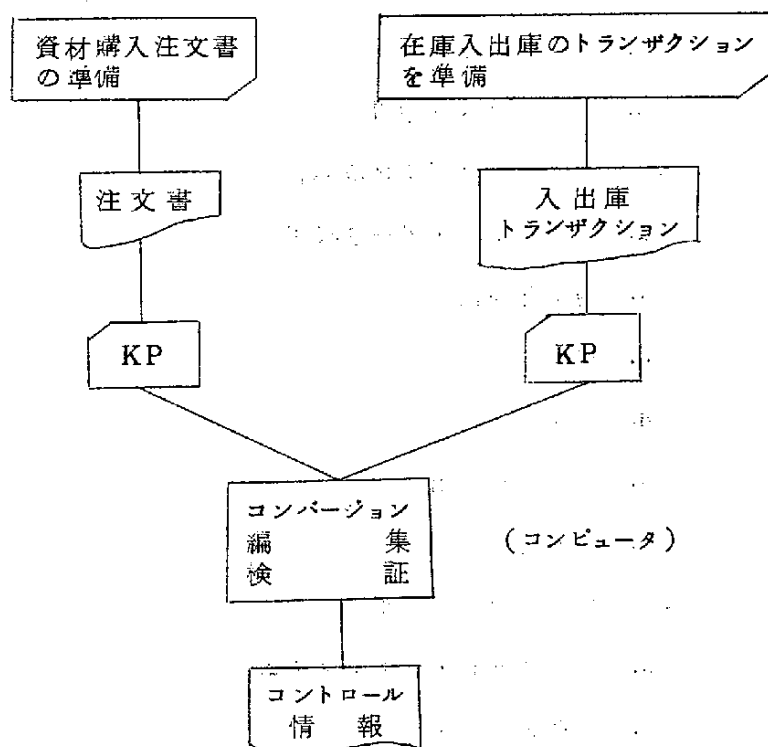
- b. アウトプット "

- c. ルーチン・プロセスング "

- d. エクセブション "

- e. コントロール "

〔インプット・サブシステムの例〕



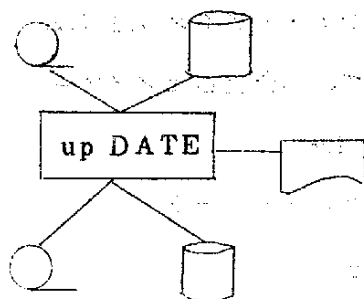
この例では、インプットがカードになっているが、代りに紙テープやCRTを使ったらどうか等を考える。合理的なオペレーション、効率的なコンピュータ利用を考えながら設計するわけである。

㊦ プログラム仕様書

フローチャートに基づいて、その中の夫々のプロセス毎に下記を記述する。

- a. プログラム概要
- b. マイクロ・フローチャート

【例】「up DATE」のデータ・フロー・チャート（図 10）



c. 論理フロー

（1）論理フローを表すには 3つの方法がある。

一つは 叙述的表現

二つは、ブロック・ダイアグラム

三つは デシジョン・テーブル

叙述的表現は文章力を要求されるので、よい方法ではない。

ブロック・ダイアグラムも限られた枠の中で書くので誤解がおきやすい。

デシジョン・テーブルが最良の方法である。

米国ではデシジョン・テーブルが最良の方法と判っているが、伝統的な方法～ブロック・ダイアグラム～を打破る事に抵抗があって、採用している企業は少ないのが現状である。

d. インプット仕様書

e. アウトプット //

f. ファイル //

以上 a～f をプログラム毎に 1つのファイルにまとめてマスターを作る。

(7) システム・スペシフィケーション (システム仕様書)

これは、システム・デザインの最終段階であって、それぞれの個々の仕様書をまとめて、総合化したシステム仕様書として文書化する。

(イ) システム記述書

(ロ) システム・フローチャート

(ハ) 報告書仕様書

(ニ) インプット仕様書

(ホ) ファイル仕様書

(ヘ) データ・エレメント用語集 (Data Element Glossary)

(ト) 各プロセス仕様書

(イ)～(ハ)はシステム全体を総括的にとられたものであり、(ト)は個々のプロセス仕様を集めたものである。

なお、以上の外に次の2つの仕様書を必要とする場合がある。

(チ) システム・テスト仕様書

テストのアプローチの方法を概略記すもので、実際のシステムテストはプログラムが終った段階でもう一度テスト・プランを詳細に立てねばならない。

(9) コンバージョン仕様書

概略で良い、どのような方法でコンバージョンを行なうか。

(8) プログラミング (Programming) の仕事

プログラマーの仕事は、プログラムの作成、テスト、保守、改善などである。

Task	Products
(1) スペックの理解	ブロック・ダイアグラム、 デシジョン・テーブル
(2) Coding and Compile	ソース・リスト (エラーのないもの)
(3) Testing Planing	テスト・プラン
(4) Test Dataの作成	テスト・データ
(5) Program Testing	テスト結果
(6) Integrated Testing Plan Data作成 Testing	テスト結果 テスト・プラン テスト・データ テスト結果
(7) Documentation	プログラム・ドキュメント

プログラマー、シニア・プログラマーの区別は、図示すると次表の如くである。

	ジュニア	中 級	シニア
仕 様 書 の 理 解 が 出 来 る		○	○
コーディング、コンパイラ	○	○	○
テスト・データ準備		○	○
プログラム・テスト	○	○	○
インテグレイテッド・テスト			○
文 書 化		○	○

米国では、プログラマについては上記資格に応じ給与水準が区別されている。ジュニア 8 千 \$ / 年、中級 12 ~ 14 千 \$、シニア 14 ~ 22 千 \$ である。

(9) システム・テスト

これはシステム仕様書、システム・テスト仕様書、コンバージョン仕様書そのものをテストする為のフェーズである。

(i) (7)の(e)、(f)で書いた概略計画を詳細化する

(ii) テスト・データの準備

誰が、どこからデータをもってきて、どういう方法でテストするかを決める。

テスト・データには、Simulation によるものと Real プラス Special Case によるものと 2 つの方法があるが、今までの経験からすると Real を使う方がよいと云える。Real の場合特に例外処理がカバーできないので、それに Special Case として例外データを作成して追加する。

(iii) テスト結果の検証

システム・テストの結果に対して責任をもつのはユーザであり、アナリストはテストの計画や実行を担当するものである。

従って、システム・テストにユーザを参加させねばならないし、又、結果についてユーザの承認をとる様にすべきである。

(10) ユーザ・トレーニング

(i) コンバージョン

(ii) コンバージョンの内容

○ データ・ファイルの成生……古いシステムのファイルが新しいファ

（イ）新システムのフォーマイルを作る。

○新システムのすべての帳票類の準備

○機器の操作方法の習得

（ロ）コンバージョンの方法

○Immediate(Total)Conversion

緊急を要したり、大きなコンバージョンはこの方法がよい。

One time Conversion ともいい、一度にやってしまう方法である。

○Phased Conversion

段階を追って徐々にやる方法

上記のどちらの方法でコンバージョンを行なうかはシステム仕様書に

もどり、つぎの事項を考慮して決めなければならない。

○システムのタイプ（日次か、月次か、資材のシステムか会計のシステムか等）

○データの量

○コンバージョンに対する有効性(availability)

○データの正確性

○コントロール

（12）文書化

システムが導入された後、システムが正常に動作していることを確認する。

システムが正常に動作していることを確認したら、システムが正常に動作していることを確認する。

システムが正常に動作していることを確認したら、システムが正常に動作していることを確認する。

システムが正常に動作していることを確認したら、システムが正常に動作していることを確認する。

第3章 システム・サーベイ

1. データ収集

データ・ギャザリングとは、現在稼働している業務環境の中から、必要かつ適切なデータ（インフォメーション）を選び出す作業である。

このタスクは、システム・デザインのサイクルの中で2回に分けておこなわれると考えてよい。最初はシステム・サーベイの段階で手がけられるが、これはデータ・ギャザリングの全タスクからみて、せいぜい20%程度のウェイトをおいたものとみてよい。本格的な作業はディテール・データ・ギャザリングの段階において手がけられ、ここでは組織末端の人々との接触まで含めた幅広い、まろ細かい作業が必要となってくる。

データ・ギャザリングの目的は、こうした作業を通じて次のような諸点を追求することにある。

- 現在稼働しているシステムの実態を掌握し、定義する。
- 新しい要請が何であることを確認する。
- 多くのデータの中から関係あるインフォメーションを選び出す。
- マネジメントからの要請が達成可能なものか、あるいは修正を必要とするものであるか等について評価、検討する。
- インフォメーションを組織化し、オーガナイズする。

データ・ギャザリングにはいろいろな方法があるが、以下にこれらの中から6つの方法を選び、その各々のテクニックその他について述べる。

(1) 文書の収集

文書の収集はインタビューによるデータ収集と平行して行なわれる。

本作業に際しては、前もってどのようなデータを集めておいて欲しい

か当該部門に伝えておくのが良い。

次に収集の対象物としては次のようなものがある。

- 組織図 (Organization Chart)
- 予算 (User Budget)
- 手続書、マニュアル (Procedure, Policy Manual)

仕事は往々にしてマニュアル通りにおこなわれていないケースが多い。マニュアルをバイブルと考えてはいけない。

- レポート (Report)
- コピーをとる。実際に使用されているものを集める。

- 様式 (Form)
- これを実際に記入したものがよい。

- ファイル (File)

(2) 現場の観察

現場に実際に出かけていって観察しながらデータの収集をおこなう方法である。

利点としては、

- 仕事の流れを知るだけでなく、例外的なケースについてはどう処理するかといった様子が知れる。
- 仕事の中断の様子（例えば電話などで）が見れる。
- アナリストと現場の人との間に直接の触れ合いがあるため以後の仕事にも好ましい関係を育成できる。

ただ、この方法は残念ながら余り利用されていない。その理由としては

- 時間や費用がかかる。

○ アナリスト側に自信がないため避けるケースがある。

観察を行なう際には、予め予備的な作業として、次のことをおこなっておくのがよい。

- インフォメーションを集めた場合の様式化を考えておく。

(Format of Summary)

- あらかじめマネジメントの了解をとっておくこと。

(3) 業務概要の掌握

この作業を遂行するためには、ユーザ側の協力が不可欠である。

各部門において、1～2名は必ず部門全体の業務概要について掌握している人がいる (Key Staff)。こういう人をまず見つけ出し、この人を介してデータの収集をおこなう。

この方法は、形式だったプロセデュアがない場合に適用されるが、特に次のような点に注意を払うこと。

- Simple Formatを準備する。
- アウトライン、短い説明文をつけることを要求する。
- 短時間に回収する。

この作業には、ある程度のインフォメーションを集め、知識ができてからでないと無理であり、アナリストとしては十分な準備を要する (システム・アナリストだけでなく、コンサルタントにも使える)。また、この作業を通じて、説明者側が問題点や解決点を指摘してくれるケースが非常に多い。

(4) 文献調査

何度も同じ作業をくり返すことを避けるために、同じようなケースを他所から探し出してきて参考にする。

では、どういうところから調査するが。B社は必ずしも大規模なソフトウェア・コンピュータ会社を通じて、必ずしも競走相手の会社から、というわけでもない。

- ソフトウェア会社を通じて (1)
- 競走相手の会社から (2) (3) (4) (5) (6) (7) (8) (9) (10) (11) (12) (13) (14) (15) (16) (17) (18) (19) (20) (21) (22) (23) (24) (25) (26) (27) (28) (29) (30) (31) (32) (33) (34) (35) (36) (37) (38) (39) (40) (41) (42) (43) (44) (45) (46) (47) (48) (49) (50) (51) (52) (53) (54) (55) (56) (57) (58) (59) (60) (61) (62) (63) (64) (65) (66) (67) (68) (69) (70) (71) (72) (73) (74) (75) (76) (77) (78) (79) (80) (81) (82) (83) (84) (85) (86) (87) (88) (89) (90) (91) (92) (93) (94) (95) (96) (97) (98) (99) (100)

ただし、この場合に他所で開発されたものをそのまま真似するのではなく、どのような問題に対し、これをどのように処理しようとしているかを調べる。

- 技術専門図書館、国会図書館から (1) (2) (3) (4) (5) (6) (7) (8) (9) (10) (11) (12) (13) (14) (15) (16) (17) (18) (19) (20) (21) (22) (23) (24) (25) (26) (27) (28) (29) (30) (31) (32) (33) (34) (35) (36) (37) (38) (39) (40) (41) (42) (43) (44) (45) (46) (47) (48) (49) (50) (51) (52) (53) (54) (55) (56) (57) (58) (59) (60) (61) (62) (63) (64) (65) (66) (67) (68) (69) (70) (71) (72) (73) (74) (75) (76) (77) (78) (79) (80) (81) (82) (83) (84) (85) (86) (87) (88) (89) (90) (91) (92) (93) (94) (95) (96) (97) (98) (99) (100)

この方法は特に統計的な情報の収集 (Statistical Information) には適している。短期間に多くの人に接触でき、又、地理的に調査対象が広がっている場合にも、比較的安い費用で調査ができるという利点をもっている。

一方、短所もある。例えば、質問の内容があいまいだと回答の中味も疑わしいし、また回答が真実かどうかを確かめようがない。

また、回答率が低い場合はサンプルとして使えないし、集められた回答に対し、アナリストが回答分析力を持たないと意味のある回答を引出し得ない。

使い方がむづかしいのでやたらに使うべきではないが、マーケット・リサーチ等には向いている。

質問状のタイプには次の3つがある。

- Multiple Choice (いくつかの中から1つを選択させる)
- Yes, No (いずれかの解答を求める)
- Short Answer (数字、量などで表現できるようにする)

質問状について必要な事項を、次のように示す。

- 質問の説明書にどのように解答するかをはっきりと説明すること。
- 質問事項はわかりやすく短かくすること、回答時間も短くて済むようにする。
- 回答〆切の時期を明示する。
- どういうところに配布したかを明らかにすること。

(6) 面接（インタビュー）

面接はデータ収集の方法として最も良く使われているが、同時に最もまちがって使われているケースが多い。

特にアナリストが準備不十分なままインタビューにのぞんだり、インタビューの結果を十分にドキュメントできない場合には、そのデータは客観的なものとして取り扱うことはできない。

こうした事態を避けるために、インタビューを十分に組織だてて準備する必要がある。

(1) インタビューには次のタスクがある。

a. インタビューの準備

- インタビュー・ガイドを作成する
- インタビュー・スケジュールを作成する
- インタビューの結果をどう処理するかを決めておく

b. インタビューの実施

c. インタビューの結果をドキュメント化する

(2) インタビュー・ガイドについて

＜第1段階＞ Listing of Topics

まず、インタビューを始める前にどういふことを質問するかというトピックを盛り込んだものを作成し、あらかじめ相手方に配っておく。(同時にインタビューのスケジュールを知らせておくのがよい)

＜第2段階＞ Specific Questions

それぞれ質問の相手によって質問の内容は変わってくる。例えばシニア・マネジメントとオペレーティング・マネジメント、また事務的なオペレーションに関係している人とでは、それぞれ異った質問を準備しなければならない。

では、どんなことが質問されなければならないかについて、次に1つの例をリスト・アップしてみる。

シニア・マネジメントに対して

○ マネジメント・ポリシー

- 将来の計画
- 予算……………etc

オペレーティング・マネジメントに対して

- 組織
- マネジメント上の問題点
- 彼等の関与している予算
- 必要としている情報は何か……………etc

直接作業要員に対して

- 自分たちの Procedures (Word)
- どのような手続を使っているか
- トランザクション、ファイル、アウトプット
- 他のプロセスとの相互関係

○ 彼等が考えている問題点、それに対する解決策………… etc

(イ) インタビューのスケジュール

インタビューをどのような順序で進めるかを決める。

例えば1つの方法として、トップ・マネジメントから順に質問相手を下ろしていく方法がある。この方法では、次の誰れに会って話を聞くのが最も効率的かを上役から聞き出すこともできる。

(ロ) インタビューの実施

インタビューの実施に際しては、特に次の諸点に注意することが必要である。

- 何が目的でインタビューをするのかを相手によく伝える
- インタビューはあらかじめ準備した質問によって進める
- ノートをとる（テープをとる。）ときは相手の許可をとる
- シニア、ミドルのマネジメントはできるだけその部屋でおこなう
- 余り積極的すぎても困る（積極的なのは好ましいが）
- 相手側に話させる
- インタビュアーは何でも知っているという態度は避ける
- 回答が不明確なときは聞き返す
- 政治的な問題（例えば企業内の）にはタッチするな
- 他所での回答と異った答が出てきても責めてはいけない
- 回答に食い違いや矛盾があったら第3者を通じて調整せよ
- 回答は強制するな
- 相手から更に有力な情報の聞き出し手を聞き出す
- 専門語を余り使わない（相手側の専門語は理解するように）

- あとで追加して聞けるようにつないでおく
- 引きぎわを上手に

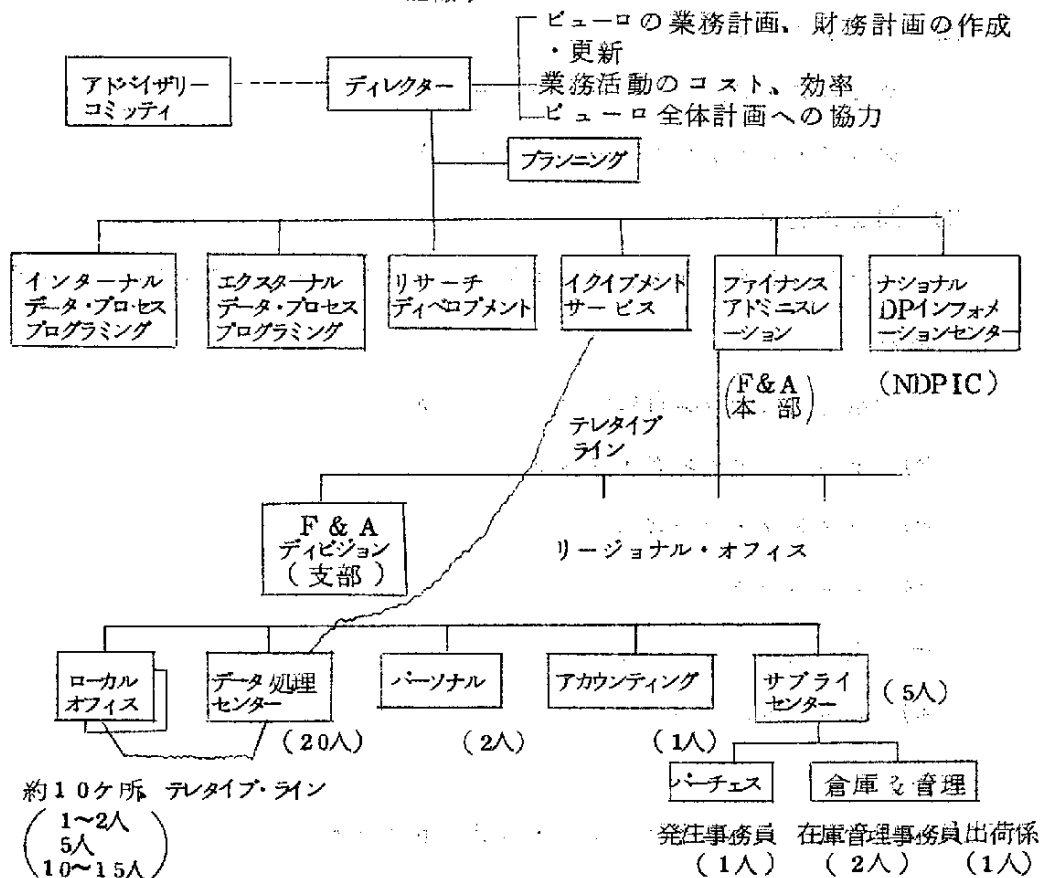
インタビューについてのワークショップ

インタビューのワークショップは下図に示す様なある組織体を想定し、「現行の事務体系を機械化」せよと言う課題で出発する。

インタビュー・ワークショップの狙いは次の2点にある。

- (1) インタビューする方の仕事の分担をどうきめるか
- (2) インタビュー・ガイドをどのように作成するか

(BIGDPCの全体組織)



2. データの分析

(1) 目 的

データ分析の目的は次の3つである。

(イ) データを体系化すること

(ロ) 現行のシステムを記述出来るようにすること

記述の方法としては、つぎのものがある。

○ フロー・チャート

○ デシジョン・テーブル

○ 文書記述

(ハ) 問題の定義

(2) アナリストの分担

ジュニア・アナリスト……………データ収集

シニア・アナリスト……………データ分析

(3) 問題の定義

アナリストは、基本的問題は何かについて充分見分けることが大切である。

問題定義は次の項目についてなされる。

(イ) オペレーティング・コスト

(ロ) カストマ・サービス

(ハ) キャパシティ

(ニ) 組 織

(ホ) システムの効率

3. デザイン・リクワイアメント・ステートメント

の D・R・S はシステム・サーベイの中では最も重要な文書である。

これまでの問題の定義、システム・サーベイ、データ収集、データ分析の各タスクより得た情報をもとにして、以下の事項についてマネジメントとユーザ向けに報告書を作成することである。

- (1) インフォメーション・フロー …… 情報の流れを明らかにする。
- (2) 目的 …… 新システムの目的を明らかにする。
- (3) 制約条件 …… 新システムの制約条件を明らかにする。
- (4) 仮定 …… 新システムの仮定を明らかにする。
- (5) インプット …… 新システムのインプットを明らかにする。
- (6) アウトプット …… 新システムのアウトプットを明らかにする。

(1) インフォメーション・フロー …… 情報の流れ

現行システムにおける情報の流れを作成する事により、現行システムの問題点、新システムの目的が明らかにされねばならない。

(2) 目的 …… 新システムの目的を明らかにする。

最小限次の 3 つを明らかにせねばならない。

- (イ) ユーザ・サービス …… ユーザの要求に迅速に対応すること。
- (ロ) マネジメント・コントロール …… 在庫管理の効率化。
- (ハ) 事務量の軽減 …… 在庫管理の事務量の軽減。

在庫管理システムに事例をとれば

- (イ)については……ユーザ側の部品の調達時間の短縮。
 - (ロ) …… 在庫コントロール、予算コントロール。
 - (ハ) …… 在庫事務の合理化
- などであろう。

(3) 制約条件 …… 新システムの制約条件を明らかにする。

システムには制約条件がつきものである。これを明確にしておかないと、あとあと“何故それをやらなかった”と言うトラブルが生ずる。具体的には制約条件とは、アナリストのコントロールにない条件、状況、方針などでシステム・デザインの場合の限界点のことである。

例えば、ハードウェアの点でコンピュータ・センタが分散していることとか、法的枠、その他競争状態がシステムに「制約」を与える事が極めて多い。

(4) 仮定

情報が不十分な場合、或は全体的な承認が得られぬ場合「仮定」と言う形が設けられる。

不明確な条件はどんなものでもリストし、マネジメントにより解決されねばならない。そして、不確実性を解決しないうちはシステムの開発を進める事は出来ない。

例えば、オンラインか非オンラインか等がこれに当る。

(5) インプット

主なインプットは示されねばならない。

例えば、請求伝票とか、納品書

(6) アウトプット

主なアウトプットは示されねばならない。

例えば、在庫品のカタログ、在庫リスト、注文書

なお、インプット、アウトプット共システムの範囲を規定するのに役立つ事は言うまでもない。

D . R . S のワークショップ

D・R・Sのワークショップは、先の「BIGDPC」における補給品在庫管理システムを、これまでのシステム・サーベイ、データ・ギャザリングで得た情報を基にして、サービス、コントロールの見地から立案せよ……と言う課題で出発する。

システム・サーベイの最終段階である。

4. システム・デザイン・プロポーザル (提案書) の作成

これはシステム・サーベイの最終段階である。

(1) プロポーザルの準備

(イ) まずいくつかの代案を作る

(ロ) (イ)の中から半分位を第一次選抜する

(ハ) リジェクトした代案についても事由を簡記する

(ニ) 選ばれた個々の案について詳細展開する

(ホ) 各代案を比較し易い様に記述する

(ヘ) レコメンデーション

どちらかと言うとプロジェクト・チームとしてどの案が良いと言う様なきめつけはしない方が良い。なぜなら最終決定は user が行うからである。

プロポーザルは、一般的には客観的に叙述し、一見どの案が良いのか判らぬ書き方がベターと言われている。

(2) プロポーザルに記述する内容

(イ) デザイン記述

○ フロー・チャート

プロセッシング・アウトプット、インプットを記述する

○ モジュール化

- 各モジュールの範囲を明確にする
- アウトプット
- インプット
- データベース

媒体は何か、構成、を記述する

以上によって、経済性、技術的可能性、オペレーショナルな可能性が明確化される。

(ロ) システムに考慮すべき諸要件

- 人 何人位要するか、特殊なジョブはないか
- 物（機器）
- 手続 どの様なオペレーションになるのか
- 組織 組織の変更があるのか
- コンバージョン 現行→新システムにコンバージョンするには一
気にやるのか、漸進的に

(ハ) 費用と効果

- 開発コスト
- オペレーショナルなコスト 回線料、端末コスト、メン
テナンスに要する人、入出
力チェッカー
マシン・オペレータ
 - 目に見えぬメリット、ディメリット モラル、インモラル

(ニ) スケジュール

(3) 代案間の比較

(イ) 代案のエレメント

ア) システムのタイプは、マニュアルか、オンラインか、オフラインか

○ ファイル媒体

○ オペレーション

電 信 A

(ロ) 各代案が充足出来る要件と不可能な要件

(ハ) 評価の基準

○ 開発コスト

○ オペレーション・コスト

○ 開発時間

○ 運用の複雑性

○ 利 益

○ 新 設 備

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップはつぎの概要のような会社を想定し、この会社について事務処理の機械化をデザインせよと言う課題が与えられる。(次ページ)

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

システム・デザイン・ワークショップ

※ アメリカ・コンシューマー・サービス会社について

A. 沿革

アメリカ・コンシューマー・サービス会社（A.C.S社）は1946年、質の高い小売組織として設立された。戦時中衣類製造業として多額の金を蓄積した5人の兄弟は、ボストン、ニューヨーク、フィラデルフィアに、増加しつつある中産階級へ・特選・高級品を販売するために小売店を開設した。1949年までにA.C.S社は製造設備を売り払い、東海岸と西海岸に店舗を拡張し、25店舗までになった。1952年、A.C.S社は米国のあらゆる主要都市に店舗を持った。しかしながら、1967年までにかなりの数の吸収、合併（A.C.S社は2つの小規模な小売チェーンを合併し、いくつかのディスカウント・ストアを吸収し、又ジョージア州にある繊維製造工場を買収した。）によって特色づけられる1.5年間の急激な成長の後、A.C.S社は50州に合計794店舗を所有することになった。現在の拡張率は今後5年間は続くと予想される——すなわち、A.C.S社は米国に1年間におよそ10の新店舗を開設しつつあるということになる。ヨーロッパの子会社の吸収を現在考慮中であり、自動車リースのチェーンの買収といったような、多くの他の新しい投機が目下行なわれている。A.C.S社の経営者は現時点を現行の運営手続きの統合と改善の時期とみなしている。経営者は次のように述べた。「今後の拡張に取りかかる前に、現在の組織の“完全な集中管理”を果たしている、ということが保障されねばならない。」「例えそれが“より多くのコンピュータや関連機器の導入を意味するものであっても、このような管理を達成するのに必要なあらゆるステップをとる”にやぶさかでない。」

B. データ処理活動

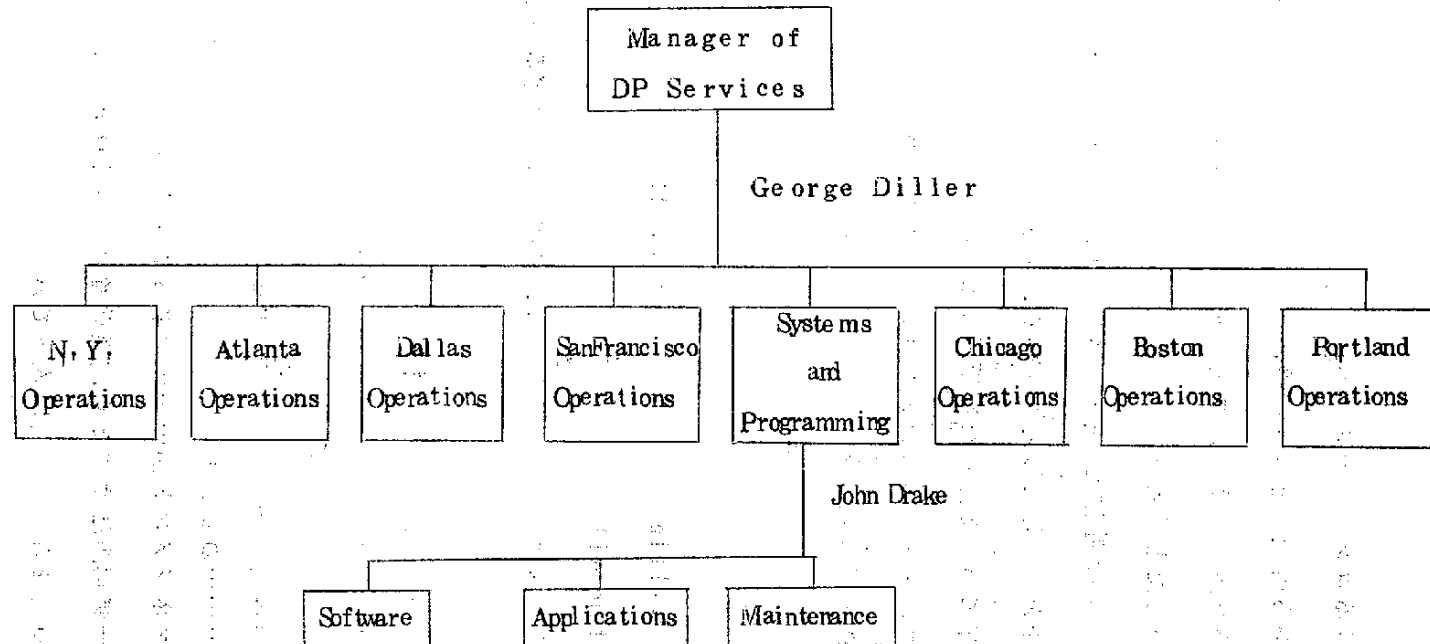
1949年、ACS社は在庫記録を維持するのに役立ついくつかの電子会計機を賃借した。又EAMの装置が得意先の会計レコード・キーピングのために使われた。1952年までに、全ての主要地方事務所がEAMの装置を所有した。そしていくつかのより小規模な地方事務所はサービス・ビューローを利用していた。1961年、ACS社が当時財産管理を受けていた会社(Halaby and Company)から小グループの小売店を吸収した際、ACS社は当社の最初のコンピュータ〜小規模な第2世代の商業用コンピュータ〜を導入した。1962年の始め、ニューヨークに小規模のデータ処理スタッフが公式に組織され、そして数ヶ月以内にシカゴ、アトランタ、ダラス、サンフランシスコに4つの地方コンピューティング・センタが設立された。

各地方コンピューティング・センタはニューヨークでの最初の導入と全く同一の機種構成を保持していた。最初のシステムの殆んど全てがプログラミング請負機関によって完成された。その機関の活動はACS社のDPSサービス部により監督されていたし、その機関は現在でも全てのデータ処理システムの開発、設置、保守の責任を負っている。しかしながら、データ処理サービス・スタッフは現在、小売店の諸活動にだけ参画している。すなわちこの時点ではディスカウント業務とか製造のための作業はしていない。

1966年の初めに、トップ・マネジメントは第3世代のコンピュータを導入するという意思決定をした。1966年の第3四半期に、ニューヨークに大規模なコンピュータが導入され、3ヶ月以内にさらに6セットのコンピュータが導入された。そのうち4セットは4つの現存の地方コンピ

ュータ・センタに、2セットはボストンと、ポートランドの新センタに導入された。これらのコンピュータは第2世代のハードウェアと置換されたけれども、ある不確定の期間、第2世代の装置と平行して運用するということが計画された。6つの地方に導入されたものは、何らのコミュニケーション装置によっても、お互いにも、又はニューヨークとも連結されていない。各地方センタは、経営者が現在の状態に対して経済的に実現可能な代替案が存在するということを確認するまでは、それらの地方のサービス・ビューロとして機能し続けるだろう。ACS社が彼等の新しいコンピュータを発注した時、その発注は、もともと製造会社の要員により実施され、ACS社の経営者の文書で提出されたシステム・サーベイに基づいてなされた。サーベイ・チームの主要メンバーは、DPサービス部マネージャシステム・プログラミング部マネージャおよび2人のプログラマであった。システム・プログラミング部マネージャ以外の全員が、ACS社の最初のコンピュータと一諸に入手したスタッフの中核であった。現在のデータ処理サービス部のスタッフは表1の組織図に示されている。その図に含まれているのは要員名とその上に附記してある要員の作業の記述である。

表 1 : A C S 社データ処理サービス部組織図



※ コンピュータ・システムのコストと利益についてのコメント

米国におけるコンピュータリゼーションは、初期の時代にはコスト軽減、他社がやるから当社もと言う競争、自動化の為に自動化、経営者に情報提供とか……どちらかと言うと、コンピュータ化のベネフィットに対してはあいまいさが多かったと言える。

しかしながら最近では、減るコストより、増えるコストの認識、即ち会社の将来の展開を予測してのとりかかり、又はオートメ化の目的をはっきりさせると言う様に健全な方向をとる様になった。

従って、コストと利益の考え方がシビアにありつつあり、特に利益については、EDP部門がきめるのではなく、ユーザ部門がきめるという考え方になっている。

☆即ち

Are you willing to pay \$ ——— for the benefits which the new system will produce?

そして、新システムの利益について、見体的にはつぎの事項がポイントになる。

- ※ ……が出来る
- ※ ……になる
- ※ 処理時間が………になる
- ※ 顧客数が………だけ増加する
- ※ ……の質が上がる
- ※ マネジメントが………出来るようになる

なお、ユーザ部門だけでなく、経営者自体がEDP化について深い関心と認識をもたねばならない時代に入っている。

第4章 プロジェクト・マネジメント

1. プロジェクト・マネジメント (Project Management)

次の4つのトピックスをとりあげる。

- 1) 前提条件
- 2) プロジェクト・プランニング
- 3) プロジェクト・コントロール
- 4) プロジェクトの完了

この中で2)と3)は、プロジェクト・マネジメントのMajor Functionといえる。

1) 前提条件 (Prerequisites)

(1) プロジェクト・マネジメントの規定

プロジェクト

関係する諸活動の集積であり、一定期間のもとで特定の結果をアウトプットするもの。

プロジェクト・マネジメント

人をリソースとして「活動」を管理すること、従って、人の管理と云う側面は有しない。

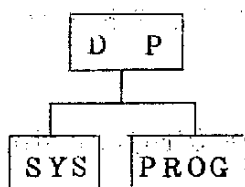
この事から労務管理の視点と活動管理の視点とが、プロジェクトの内部で葛藤を呈することがある。

(2) スタンダード

システム開発プロセスにおける標準を指す。これには、メソッドとパフォーマンスの二面から考えられる。

(3) DPの組織

(イ) Aタイプ



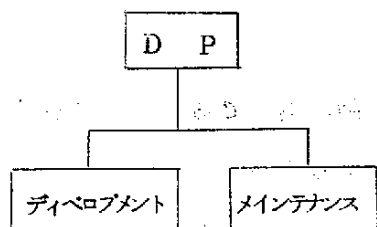
このタイプは、現今米国で最も多く見られる組織である。しかしながら、プロジェクト・マネジメントの観点からすると、

- 「Development」と「Maintenance」の区分が不明確となり易いし、

システム・アナリストとプログラムのグループ・リーダーが、プロジェクト・リーダーのもとに一元化しにくい、

と云う欠点を持つ。

(ロ) Bタイプ



このタイプも比較的多いが、Aタイプと同様な観点からすると、

- 必要なあらゆる開発リソースをディベロプメントに集中する試みが、長所といえるが、ディベロプメントが成長して、多人数になると、

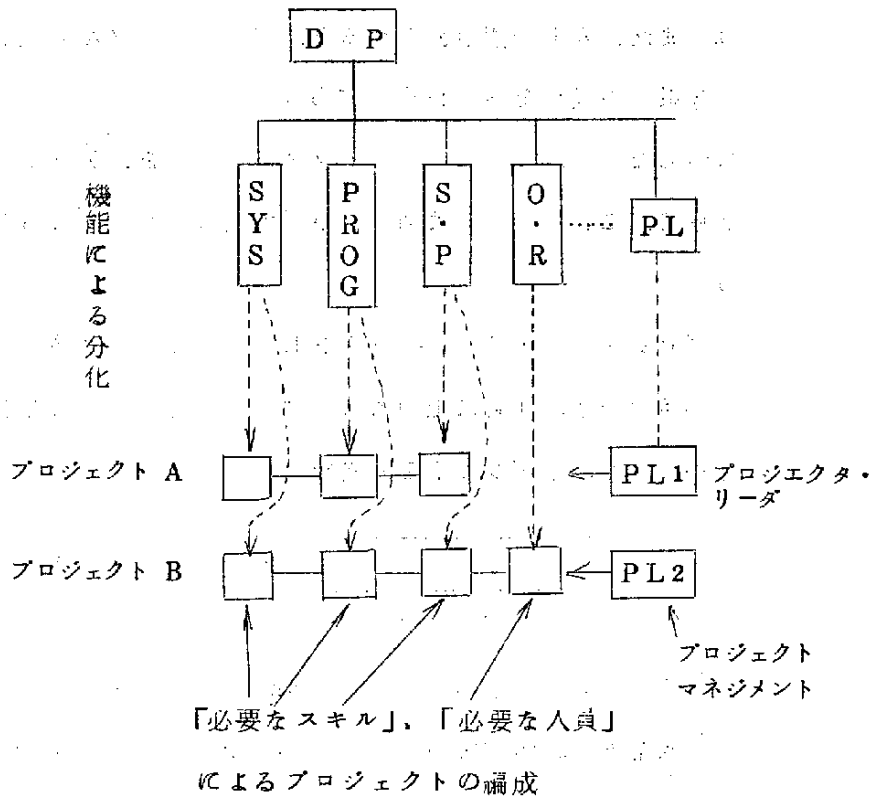
しばしばサブ・グループに、主としてアプリケーションを基準に分化し、硬直化する懸念を有している。

こうなった場合

- 組織上のバランスに対する配慮が仕事量に対する適正人員

化を妨げるという弊害を生ずる。→マトリックス型組織

Ⅳ Cタイプ



このプロジェクトが複数存在する事を考えれば

- 組織上の縦の訓練
- プロジェクト上の横のコントロールおよびダイナミックな要員編成

という利点が追求出来る。→マトリックス型組織

(Matrix Organization)

○ このマトリックス型組織の姿が最も良いと思われるが、つぎのような問題点がある。

1. 報告の相手がダイナミックに変化することを嫌う。(誰に忠誠

心を奮うのか?)

2. 人間をリソースとしてしか扱わない。
3. また、人事的観点からすると、ダイナミックな変化が縦の指揮命令系統の混乱を招くおそれがある。

しかしながら、プロジェクト・マネジメントを、うまくやろうとするならば、これらの問題は何らかの形で克服されねばならないものである。

○ プロジェクト・リーダー(以下PL)とライン・マネジャの対比

共通機能は、「共に上司に報告する義務を有すること」を除けばほとんどなく、下記のような差異を有する。

	PL	ラインマネジャ
仕事の与えられ方	有限の任務	継続的任務
命令系統(部下への)	Non-official	official
人を使って仕事するや方	説得とだめ	ムチとアメ
権限(昇給、評価 etc)	無	有

このために、通常プロジェクト・リーダーはプロジェクト・マネジャの権限の裏付けを持つ場合が多い。(PMは、そうする配慮が必要)

ここでは、プロジェクト・リーダー……………
プロジェクト・マネジャ……………}として考えている。

(4) マネジメントへのコミットメント (Management Commitment)

この場合のマネジメントとは、DP部門のマネジメント、及び全社的なシニア・マネジメントを指す。

一般にマネジャは、ターゲットとリソースの関係を認識せず、ター

ゲットのみを指示する事が多い。

我々としては、先ず確実にリソースの要求を見積り、その確保のもとにターゲットが守られるという事を、しっかり認識してもらう必要もある。

そして、そのリソース確保の承認と、場合によっては、そのリソース確保をマネジメントに認識してもらうべく、働きかけねばならない。一般に、マネジメントとの間に、プロジェクト・マネジメント/プロジェクト・コントロールという事に関する話し合いが行なわれていないといえる。

(5) リソース・コミットメント・チャート (Resource Commitment Chart)

使用済または使用中のリソース、リザーブしたリソースを各プロジェクトで統計する。→これを Loading Curveという。

Loading Curveはわるく共1ヶ月に1回は Update されるべきで内容としては、先行3ヶ月程は持つ必要がある。

また、プロジェクトの優先順位を決めるためにも、これは重要である。

最も重要な意味は、これによって、限られた全体リソースを各プロジェクトの優先順位のもとにバランスをとることにある。

このリソース・コミットメントの承認者はユーザでなくてはならない。

複数のプロジェクト・コントロールについては、現在、残念ながらあまり研究されておらず、早急に着手されるべき時期といえよう。しかし、この問題は DP 部門の問題ではなく、ユーザ側の問題といえよう。

2) プロジェクト・プランニング(Project Planning)

(1) システム開発プロセスが13フェーズより成る事は前に述べた。

仕事の細分化は機能の明確化の上で、大変有用なことである。しかし、プロジェクト・マネジメントの観点からは、余り細分化することは、かえって得策でない。

そこで、前に述べた13のフェーズをマイナー・フェイズとして、PlanningとApprovalを単位とする区切り方でメジャー・フェイズを設定する。

Major フェーズ	Minor フェーズ	Products
	1. Application Selection	System Def(1)
[A] Survey	2. Systems Survey	System Def(2) System Proposal
[B] Systems Analysis & Design	3. Data Gathering 4. Data Analysis 5. } 6. } Systems Design 7. }	System Def (Final)
[C] Development	8-a. Programming 8-b. Procedure Writing 8-c. Integrated program Testing 8-d. Special Plan	
[D] Implementation このフェーズはプロジェクトではなく、ユーザが主体になるべきもの	9. System Tesing 10. Conversion 11. Training 12. Documentation	

(イ) プロジェクトの定義づけ (Project Definition)

少なくとも、つぎの 3 つが付加されるべきである。

1. 組織図……………誰れに報告するか、誰れの承認を得るか
2. 概括計画……………予算、スケジュール
3. 詳細計画……………" " " " " "

他にはオペレーション経費の見積などの付加も考えられる。

(ロ) これらのプロジェクトの定義づけ、システムの定義づけ、付加的

なプロジェクトの定義づけなどを「Phase End Control

Document』とよび、これはプロジェクト・リーダーの責任で作成されるべきものである。

また、3つの関係を記述するとつぎのようになる。

システムの定義づけで記述された業務に規定されて

付加的なプロジェクトの定義づけが書かれるべきであり、

したがってシステムの定義づけの変更は、当然後者に影響を及ぼす。

プロジェクト・コントロールのワークショップ(1)

※ 問題の設定

ACS社のDP部門のSP部長John Drakeが退社するので、後任者の立場となり、その置かれている立場から見たACS社の現状認識と目的の把握、および、今後の活動計画を提出せよ。

また、すでに出来ているスタンダード・マニュアルについても考察せよ。

資料としては、邦文テキスト「プロジェクト管理の手法Ⅲ」を使用する。

本来、ターゲットを設定する（時間の規制）のは、のぞましくないが、このワークショップでは、その設定を行なって実施する。

※ ワークショップの目的

プロジェクト・リーダーがプロジェクト・プランニングに当って、環境、リソースから、いかに正当性のあるスケジュールを立てられるか。

○ A班の着目点

1. リソースの不足（人員増員）
2. プロジェクト・コントロール組織の確立

○ B班の着目点

プロジェクトの要請点の確認

1. プロジェクトのゴールは何か
2. 〃 のタスクの確認
3. リソース・時間の見積り
4. スケジューリングの実施
5. チェックポイントの確立
6. プロジェクト・コントロールの確立

○ C班の着目点

システム・デザインの不備（全面的再検討）

1. 全般的な要件の未充足
2. 経営目的に合致しない

スタンダードの内容改正

1. ユーザの参加を明確化
2. プロジェクト・コントロール面の完備

講師の論評 このワークショップの目的は、プロジェクト・リーダーの

2つの機能のうち、第1の機能に関するものである。

プロジェクト・リーダーの機能は、

1. プランニング（シニア・マネジャへの説明／観告）

2. プロジェクト・チーム・コントロール

作業結果に対する論評

◎ ACS社ではプランニングが出来ていないので、このプロジェクトが開発のどの段階にあるのかが不明確である。

◎ 従って、現段階を明確化するプランニングが必要であり、ここではサーベイ実行のための計画を立てねばならない。

◎ 経営目的を明確化し、それに矛盾しない形で給与システムを開発する必要がある。

◎ その他必要と思われる事項

○ リソースの必要性と、マネジメントの承認

○ 組織の確立（ユーザの参加、プロジェクト・チームの編成）

○ スタンドアードの改正と、それを使用すること。

(3) プロジェクト・プランニングのまとめ

(i) プランニングの5つの効果

a. 仕事・担当者・所要時間・結果・費用の明確化

b. リソースの適切な配分（適材適所）

c. チェックポイントの明確化

d. マネジメントの承認（及び改善意見）の獲得

e. マネジメントを教育する機会

プランの
提出により

(ii) プロジェクト・リーダーの機能

a. プランニング

b. コントロール

(1) プランニングのステップ

a. 準備（問題の定義）

- プロジェクトのタイプ（時間志向か、リソース志向か…）
- プランのレベル（フェーズか、タスクか…）
- プロジェクト定義づけの作成

b. フェーズ、タスク、活動の決定

- 概括計画 → フェーズ、サブ・フェーズ、段階
- 詳細計画 → タスク段階
- 個別計画 → 実際活動段階

c. 必要なスキルの決定

- 見積りのために必要
- 不足スキルの導入、訓練の必要性

d. タスク・タイム（人／日）見積り

（詳細は後記（二）参照）

e. スケジューリング及びリソースの配分

- 人／日の経過時間への換算
- リソース配分に当って、他プロジェクトへの影響も考える
- ユーザ・マネジメントによる必要リソース配分の責任の明確化

スケジューリング技術

- Target Date プランは簡単だがコントロールに使えない
- GANTT Chart Target DateとPERTの中間
- PERT コントロールには良いがプランに使っていく

f. チェックポイントの確立

チェックポイントの目的

○品質管理

① プロジェクト・リーダーのチェック（業務毎に）

② DPのチェック（プロジェクト外のDP担当者のテクニカル・チェック）

③ ユーザのチェック（アプリケーション面）

④ マネジメントのチェック（財政面）

○進捗状況のチェック

事後的チェックになるので余り意味がない。

g. 予 算 化

一般には概括計画で行なう。

h. 提 出

マネジメントの検閲と承認を求める。

(二) 見 積 り

a. 見積りの方法

○ タスクの詳細を知る。

○ 請細見積りは、全ての条件が明確化してから行なう。
（実際上はなかなかむずかしい）

○ 見積りは控え目であること。

（控え目に見積り、仕事はキチンと行なう）

○ 見積りに当り、仕事の量・複雑性の予測経験者が必要。

（見積りは、前もって算出されている単位時間と、仕事の量、
複雑性により計算することができる）

b. 見積りの技術

- 経験的
- 直観的
- 数値的
- パフォーマンス・スタンダード

c. 見積りの時点及び内容

	概括計画	詳細計画
① サーベイ終了時	② ③ ④	②
② デザイン "	③ ④	③
③ 開 発 "		④
④ 施 行 "		

d. 見積りの正確度（一例）

	②の見積り	③の見積り	④の見積り
① サーベイ終了時	±10%	±20%	±30%
② デザイン "	—	±10%	±20%
③ 開 発 "	—	—	±10%

プロジェクト・コントロールのワークショップ(2)

※ 問題の設定

ワークショップ(1)でのプロジェクト・スケジュールとして、もっと正当性のある Target date が与えられた。

プログラミングについても、個々に条件書の出来上り、難易度などの具体的な条件が与えられた。更に要員の数、スキル、経験も明示された。この上でマネジメントは、プロジェクト・コントロール・グループ、リソー

ス・コミットメント・プラン、ゴールなどの承認をしている。

以上のもとでプロジェクト・リーダー（あなた）は、各要素及び全体のプロジェクト・スケジュール、要員割当、コストなどのプランニングをし、ユーザにこのプランを説明するときのポイントを示すように課せられる。

※ 講師の論評

次の項目についてのコメントがあった。

チェック・ポイント

タスク・スケジュール

プロジェクト・リーダーのプロジェクト・ステータスのチェック

個人の責任

○プログラミング難易度のウエイト

① 分類は 3 段階

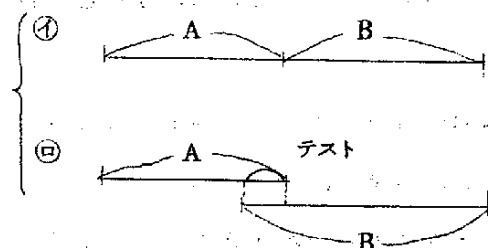
{ 困 難
普 通
簡 単

② ウエイト付

	D.	A.	S.
算術的方法	3	2	1
幾何的 "	4	2	1

幾何的ウエイト付けをとるべきである。

○プログラミング時間の見積り方



オーバーラップ

が、一般的には㊦が正当であるが慎重な立場からは㊧が良い。

3) プロジェクト・コントロール

プロジェクト・リーダーの機能

リーダーシップ/コントロール……厳しく

プランニング ……控え目に

(1) プロジェクト・スーパーバイザ

(2) 進捗状況のコントロール

(3) 適切なアクション

(4) 質のコントロールとチェックポイント

(5) プロジェクト・ログとフィードバック

(6) 活動報告/月報

(7) 例外事項報告

(8) プロジェクトの終了

完了と勧告

○ 進捗状況のコントロール

(チェックポイントに来たら見直す)

このやり方は欠点がある。即ち事後的になり、修正がきかない。し

たがって事実を知るだけになる。

実際に必要なのは問題の兆が現われたら、当初のスケジュールに

間に合う様にコントロールすることである。

① プログラム・リポートの必要性

各プログラマ毎に人/時間表を作ること。

④ 週間タイム・シート

⑤ 追加を要する時間の書込み

全員のタイム・シートを総合し、チーム全体の事態の予測を採知する。

○ 修正の方法

① 変更 予定の活動……… 予定の活動……… 変更

② 応援 予定の活動……… 予定の活動……… 応援

○ コントロールのサブ機能

① プロジェクト日記

② 月間報告 { 予算が 5 % 増加する (5)

③ 例外事項報告 { 2 ~ 3 週間経過時間が増加等々 (4)

プロジェクト・コントロールのワークショップ (3) (4)

※ ワークショップ(2)を更に発展させたものである。

即ち(2)のごとく、3月1日スタートのプロジェクト・プランを発表してすぐ後に“あなた”(2)のプロジェクト・マネジャ)は、ボスから他のデータ・プロセスの導入のサーベイを指揮するように依頼された。

……………そしそこの仕事をした……………

今8月1日で、“あなた”は先のサーベイの仕事から戻った。ボスからこのプロジェクトの状況を把握し、レポートするように課せられた。

(1) プロジェクトの状況を分析し、レポートすること。

この際、必要なコメントは下記ディスカッションの項にある。

(2) 計画されていた各アクティビティの進捗状況、ユーザの受領状況、要員の一部異動、負荷、気持、うわさなどの状況がある。

(3) 計画したときのプログラミングの各段階での作業見積基準があって、それによるプロジェクト・スケジュールがある。

- ③ プログラミング概要と詳細 (3) 質問と回答 (4) 質疑応答
- ④ コンバージョン計画 (5) 評価 (6) 報告 (7) 評価 (8) 報告
- ⑤ プロジェクト・トレーニング (9) 評価 (10) 報告
- (4) プロジェクトの実際の進行状況の概要と詳細レポートがある。これは計画と実績が対比されている。
- ⑥ ⑦ プロジェクト・マネージャはどのように状況分析し、処置をとるか？
- ケースの各要素のディスカッション
 - a. プロジェクトの状況
 - b. プロジェクト・コントロールの有効性
 - c. 次のプロジェクト・リーダーに任命されて、仕事をやって来たオマリー氏の有効性
 - d. プロジェクトの効果を増すための必要な手段
 - 講師からの諮問
 - a. 誰れを昇進させるべきか
 - b. トレイシー氏の仕事は、スケジュールより進んでいるか遅れているか
 - c. プロジェクト・チームの要員の評価は
 - d. 誰れが基本的にこの問題を引き起こしたのか
 - e. 何故プログラム(3)がスペックに合わずユーザに受け入れられないのか
 - f. プログラムの給与はどうあるべきか
- (9) 講師の論評
- a. 昇進について

計画標準値	{	実績	
		80%以内で出来た	Aクラス・プログラマ
		100%	B "
		120%のオーバーで出来た	C "

各人の仕事の能率を分析すると、リチャート氏がこの対象はなる。

b. トレイシー氏は、余裕があるのではなく、計画の-20%で作業アップするのは当然。他の遅れている仕事を計画すべきである。

c. 各人の評価、オマリー氏は訓練中のプロジェクト・リーダーという見方で、そのための計画、作業指示、育成を計るべきだ。これをもとにプロジェクト・リーダーはきちんとしていない。各プログラムの実績の評価をし、リチャート氏をCからBクラスへ上げる。

d. プロジェクトを現在のような問題ある状態にしたのは私(もとのプロジェクト・リーダー)である。

e. プログラム(3)の作業過程を分析してみると、チーフであるオマリー氏のレビューが不足している。特にフローチャート作成の時間が標準計画作業時間より短かく、少し手を抜いているのではないか。

f. 上記(a)、(c)のモチベーション。

プロジェクト・リーダーとして、プランに対してコントロール・レポート、チェックポイント、をどのように作成、設定し、レビュー技術を身につけるかが大切である。

(注) 以上がブランドン・アブライド・システムズ社によるプロジェクト・マネジメントの研修内容であるが、研修終了後、本研修団はSun Oil社、Guardian 生命保険会社、ニューヨーク貯蓄銀行、ダラス国際空港等を訪問し、プロジェクト・マネジメントについて経験交流を行なった。

なお、帰路ロサンゼルスのアナハイムで開催されたFJCC

(全米秋期電算機連合大会)にも参加した。

請求
番号

経
47-4

登録
番号

著者名

書名 アメリカのアロジエ・マゼメント

所属

帯出者氏名

貸出日

返却
予定日

返却日

所属	帯出者氏名	貸出日	返却 予定日	返却日

