

解説

DIPS-1 ソフトウェアのマネジメント*

戸田 巖** 新井 克彦**

1. はじめに

DIPSとは、Dendenkosha Information Processing Systemの略称で、電電公社がデータ通信サービス用に開発を進めている標準情報処理システムである。電気通信研究所では、1967年からDIPS計画を推進している。DIPS計画の中で最初の実用機をDIPS-1と命名している。

DIPS-1 実用化の主な特長は次の2点である。

(1) 複数企業の共同開発

DIPS-1は電気通信研究所が中心となり、日本電気、日立製作所、および富士通の3社と共同で開発した。とくに過半の努力を傾けたソフトウェアは、四者が協力して一式のソフトウェアを分担作成した。

(2) 大規模プロジェクト

1968年から1973年までの5年間に約300億円の巨費を投じた。開発に従事した技術者の数も数千人にのぼると推定される。

本稿では、上記の特長を持つDIPS-1実用化のマネジメントの考え方につき、ハードウェア、ソフトウェア共通な事項およびソフトウェア固有事項につき簡単に紹介する。

2. 共同研究

DIPS-1の実用化は、国産技術の総力結集を目標として上述の4機関が共同で実施している。

具体的には、それぞれの機関が実用化に必要な技術者を提供し、装置、プログラムの設計・製造を共同または分担して実施し、それらをDIPS-1システムとしてとりまとめる形態をとっている。その際の成果、工業所有権、ノウハウを4機関で自由に使用できるように取り決めているのが、共同研究と呼んでいる1つの理由である。

4機関の分担は、次のような原則で行なっている。

まず、DIPS-1実用化の計画、全体のとりまとめは通研が担当している。ハードウェアの場合には、設計は、4機関の技術者が共同で実施する。製造については、通研が製造設備を保有していないので、日電、日立、富士通3社に、装置単位にそれぞれの社に依頼している。主要な装置(たとえばCPU)は、3社それぞれ試作を行なった。

ソフトウェアの場合には、“設計は共同で、製造は個別”の原理によっている。ただし、ソフトウェアは、一式製造すればどのハードウェア上でも動作できるので、一式を共同で設計し、これを4分割し、それぞれ分担して製造を行ない、最後に共同で4部分を一式に統合して完成するという形式をとっている。図-1にDIPS-1ソフトウェアの製造で採用した工程区分を示す。全工程中、詳細設計から接続デバッグまでを個別作業、その他を共同作業で実施している。

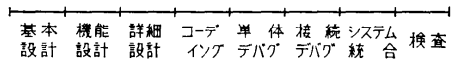


図-1 DIPS-1ソフトウェアの生産工程

共同研究開始以前は、各機関はそれぞれ独自の考え方、手法に従って計算機の開発、製造を行っていた。共同研究は、これら各社の得意とする技術を統合するという大きな利点をねらったものであるが、その実現には種々のマネジメント上の工夫を要した。

設計の時期には、4機関の技術者を一堂に集めた打

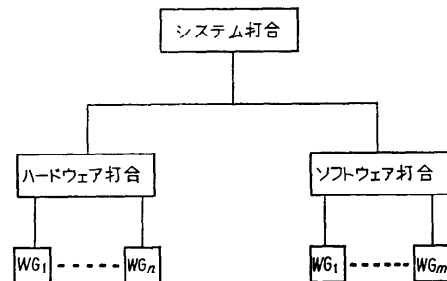


図-2 DIPS-1実用化の打合せ体系

* Management of DIPS-1 Software development by Iwao TODA and Katsuhiko ARAI (N. T. T., Yokosuka Electrical Communication Laboratory)

** 日本電信電話公社 横須賀電気通信研究所

合せを多用した。打合せは、システム、ハード、ソフトの3種に大別される(図-2 前頁参照)。ハード、ソフト関係の打合せは、さらに全体を統括する打合せ、装置またはプログラムごとの打合せ、インターフェイス、信頼度設計などの共通打合せを階層的に設置した。それぞれの打合せにおいて、通研のメンバが主査となり、打合せの計画、運営などとりまとめを担当した。

全体計画の中において、個々の打合せの役割(mission)と概略計画が上位打合せから指示される。個々の打合せでは、検討項目を洗い出して検討の詳細計画を作成し、さらに他との調整を要する項目があるときは、検討計画の同期をとる。検討項目ごとに担当者(または社)を割り当てる。項目によっては同一項目を複数者に割り当て、並行的に検討し、個人的なかつよりや検討もれを防止する。

検討結果は資料として打合せに提出し、審議、決定し、設計の結果は設計要項と呼ぶ文書にとりまとめる。設計要項が脱稿すると、関係者が集中的にレビューを行ない完成させる。設計または製造段階を通じて、設計要項よりさらに詳細な情報や確認事項等は、連絡票と呼ぶ文書に記載して相互に交換し確認を行なう。設計要項完成後の仕様変更等は、関連担当での調整完了後、所定の設計要項変更手続きにより決裁し、関係者に配布される。

共同研究における問題点は、

- (1) 参加メンバのバックグラウンドが、機関ごとに相違する。
 - (2) 相互に直接意見を交換し得る技術者の数が、計画全体に参画する要員数に比べて少ない。
- などである。計画の目標、意義を明確にして参加要員に徹底させることは、共同研究を成功させる第一の要諦である。

さらに各機関の自主性を尊重し、既成の組織を最大限に活用するためには、

- (i) 綿密な計画立案
- (ii) 打合せによる慎重審議
- (iii) 情報の文書化
- (iv) レビューの重視

などが、有効な手段であったと考えている。

3. 大規模プロジェクト

すでに述べたように、DIPS 計画は巨額の開発費、膨大な技術者を動員した大規模プロジェクトである。大規模プロジェクトであることが、共同研究を必要

とした大きな理由であり、共同研究のマネジメント上の問題点の多くは、大規模プロジェクトのそれと共通である。その意味で前章に述べたことは、共同研究ではない大規模プロジェクトにも適用できると思われる。

大規模プロジェクト運営の最大の問題は、コミュニケーションであろう。開発プロジェクトでない、すなわち手法が確立しているプロジェクトの場合には、いかに大規模であっても、伝達すべき情報の種類、タイミング等は既知であり、そのチャネルをあらかじめビルトインできる。

これに反して開発プロジェクトにおいては、

- (1) 当初は全体像が必ずしも明確でなく、いかなる情報をどこに伝達すべきかを予測できないことが多い。
 - (2) 組織が大規模で、臨機応変なコミュニケーション・チャネルの設定が困難である。
 - (3) 情報が末端に到達するまでに、多くの人を介する必要がある伝達に時間を要する。
- などの問題が発生する。

(1)の問題には、まず全体像を適確に把握しうるkey personsに情報を集中し、明確な構想を作成し、その構想を組織の各要員に伝達する。

DIPS-1 設計の初期にはkey personsは、関連するすべての打合せに出席することを義務づけて、情報の収集と疎通をはかった。相互に情報を伝達すべき要員の数の自乗に比例して、コミュニケーション・チャネルの数は増加するので、key personsの階層的配置が必要である。

情報をできるかぎり文書化して、コピーを配布することは、確実な通信手段ではあるが、資料作成に人手・時間を要すること、高価であるなどの欠点がある。しかし大規模プロジェクトは長時間にわたること、要員の転出入、担務の変更などが避けられないので、情報の文書化は必須である。

大規模なプロジェクトの第2の問題は、個人の達成意欲の満足と緊張感の持続である。プロジェクトの目標自体は壮大である。それに比較すれば、個々の要員の貢献はきわめて小さい。また、電子計算機のようなシステムでは、相互の関連が強く、個人の自由な裁量は制約を受ける。一方、開発プロジェクトの要員の平均学歴は高い。このような環境下で、個人の達成感を満足させるには、プロジェクトの意義、目標の周知徹底、ならびに個人の担務との関係を十分に理解させて

仕事を進めさせることが重要である。もちろん具体的な作業目標を与えて、仕事の進展を実感としてとらえさせる配慮も重要である。

大規模プロジェクトの第3の問題点は、組織の硬直化と官僚化である。大規模プロジェクトを能率よく運営するためには、臨機応変に組織・体制の変更ができればならない。そのための仕組みと、常に将来を見越した組織計画の用意が必要である。

多数の要員を一定の目標に動かしていくため、とくに目に見えないソフトウェアの開発には、一定の手続き、形式に従って仕事を進めていくことが必要である。このことは、反面官僚化を招きやすい。運営上十分に配慮する必要がある。

4. ソフトウェア・マネジメント

DIPS-1 実用化の過半の努力は、ソフトウェアの開発に向けられた。

大規模ソフトウェア開発における主要な問題点は、次の3点であろう。

- (1) ソフトウェア開発計画の作成
- (2) 進行管理
- (3) 品質保証

適確な開発計画の策定には、開発対象ソフトウェアの正確な把握、所要時間、工数、計算機時間等の見積りが必要である。

ソフトウェアの場合、ハードウェアのいわゆる設計に相当する対象の完全な記述が、通常は開発期間の半ばにしか完成しないこと、ユーザの要求により設計変更が度々行われる事例が多いことなどの理由で、開発対象の適確な把握はなかなか困難である。設計事例を多数収集しておくことにより、開発対象の正確な把握が可能となる。

ソフトウェアの作成は、大部分人手によらざるを得ない。プログラムの生産性は、プログラムの規模、種別およびプログラムの力量により大幅に変動する。さらに、プログラムの作成方法によっても生産性は変わってくる。

DIPS-1 ソフトウェアの開発は、共同研究で実施することとしたこと、および大規模なプロジェクトを急速に組織したため比較的经验年数の浅い技術者が多かったことから、プログラム作成の方法を規定した作業標準を作成して、標準化された手法によりプログラム作成を進めることにした。

ソフトウェアの作成の進捗は、目に見ることができ

ない。このことがソフトウェアの進捗管理をむずかしくする。作業標準では進行管理を容易にし、プログラムの目標達成を具体化するために、ソフトウェアの生産工程を明確に定義した部分工程に分割した(図-1)。各工程では、まず何らかのドキュメント、プログラムファイルなど、目に見える生産物を作成することにして、これらの中間生産物の完成を目で確かめ、レビューを実施することにより、各工程の終了およびその品質を確認できる。さらに各部分工程に数ヶ月の期間を割り当てれば、数ヶ月ごとに進捗状況を確認できる。

ソフトウェアは、多数のコンポーネントから成る。各コンポーネントであるプログラムの作成順序、他との接続手順を精密に記述した PERT 図は、その作成過程を通じて計画を詳細化できること、およびそれによって進行管理を行えることの2面の利点を有する。

ソフトウェアの工程を乱す最大の要因は、製造途中における仕様の変更である。設計時点においてユーザとの話し合いを徹底的に行ない、設計終了後の仕様変更を最少におさえることが、大規模ソフトウェア開発成功のかぎであり、プロジェクト・マネージャの最大の責務であるといっても過言でない。

ソフトウェアの品質保証は、きわめて困難な課題である。ソフトウェアの品質は、設計、製造の両面から考えなければならない。設計品質の確保は、優秀な人材の登用が最大のポイントであるが、文書管理その他支援的業務の充実、組織的な手法、レビューの徹底などが有力な手段である。製造品質の向上のためには、前述の中間生産物の品質を常に監視し、向上させること、品質保証に専念するグループを設け、チーム全員に常に品質への関心を持続させることなどが重要である。ソフトウェア品質のメジャーに性能がある。性能は設計時にプログラマが明確に目標としうる形に分解して目標設定がなされる必要がある。性能の目標達成度については、各工程ごとに得られる情報をもとに、シミュレーション等により評価し、問題点は適宜設計・製造に反映して行く必要がある。このように常時監視を行なっているにもかかわらず、大規模ソフトウェアの開発は長期にわたるので、ソフトウェア自体以外の要因(たとえばハードウェア、システムの使用条件など)により、完成期には当初設定した目標では不十分になる事例が多い。これまでの経験では、完成したソフトウェアの性能改善は、妥当な範囲であれば比較的短期間に実施できる。それ故、中間工程で発見した問題点等はそ

都度反映するよりも、問題点を集積しておき、ソフトウェアの一応の完成をまった上で、一括して措置する工程を組むことが良い結果をもたらすことが多い。このような点から、評価の道具および手法を整備して省力化をはかっておくこともマネジメント上重要である。

5. むすび

以上、電電公社、日電、日立および富士通の4機関が共同で実用化した大形オンライン用電子計算機DIPS-1の開発に用いたマネジメントの考え方の一部を紹介した。

DIPS-1の共同開発では、4機関がそれぞれの個性

を生かしながら、技術力を結集して単独の企業では企画し得ない規模の計画を完成し得た。

今後情報化社会の進展とともに、大規模社会システムなどの開発がますます盛んになると予想される。複数の企業が、共同して1つのプロジェクトを実施する機会も増加していくと考えられる。DIPS-1実用化の成果が、今後の共同開発の参考となれば幸いである。

参 考 文 献

- 1) 戸田：DIPS 開発計画をめぐって，オペレーションズ・リサーチ，pp. 21～25，1975年4月号，
(昭和50年6月23日受付)