

## 4. オペレーティング・システムとしてのコンパイラ的设计

磯田和男，稲見泰生，斉藤直之，中村康弘（原研）

### 0. いとぐち

最初に問題になるのはユーザー側でのシステム・プログラミングの限界である。これについての意見は非常にひろい範囲に分布している。次の(1)，(2)がその両極端であろう。

(1) ユーザーはアプリケーションに全力をそそぐのが本筋であって，システム・プログラムは所与のものとして利用し，不満足な点はメーカー側に指示して行なわせる。

(2) ユーザーは自分の業務を遂行するのに最善の電子計算機をえらび，アプリケーション・プログラムの処理に必要なシステム・プログラムを自力で開発するのが，メーカーから供給されたものだけにたよるのよりも，より効果的に電子計算機を利用できる方法である。

(1)の立場をとるユーザーの方が多数をしめているが，(2)の立場のユーザーもそれほど少数ではないのが現状であると思われる。しかも(1)の立場をとっているユーザーのかなり多くはそれほどしっかりした信念にもとづいているとは断言できない。あらゆる分野にわたるJOBをかかえている計算所は別として，ある一つのユーザーの中心となる適用分野にはあまりひろい範囲にわたらないのが通例である。メーカー側で作られるシステム・プログラムはほとんどかなり広範囲をカバーしどんな適用分野に応じられるようになっていくのが通例であるので，ユーザー側でのかなり限られた範囲のJOBからみると重点的な目標にずれができてくる。この点で(2)の立場というものを無視してしまうわけにはいかなくなるのではないだろうか。

原研のようなユーザーでこの表題のような計画が行なわれるようになったのも上記のようなことが一つの出発点になっている。

### 1. 現在のオペレーティング・システム

現在われわれが使用しているのはIBM7040/7044 OPERATING SYSTEM(16/32K)であってその設計原理は〔1〕-〔5〕にあるとおりで，非常にすぐれたものといえる。ところが原研では処理されるJOBの99%以上はFORTRAN IVプログラムであり，I/O部分のきわめて少ないものが大多数をしめている。その点設計者のおいたウエイトとわれわれのおくウエイトの間になりにずれがある。また全体が1.6Kベースであって，

32K のメモリーがあるときはもっと効率のよいコンパイラができるのではないかと思われる。原研ではコンパイル時間はかなり多くの比率をしめているので上の点は重要である。またシステム一般が、原研に設置されているような低速のI/O ユニットの対象にしていないうであった。

またわれわれがこのような点で討議していたときには科学計算用の言語としてはFORTRAN IV だけしかなかった。その後ALGOLのプロセッサとしてグルノーブル大学で開発されたもの、SHARE ALGOLの7040/44版、ALCOR-ILLINOIS ALGOLの7040/44版が入手できるようになったが、IBM社としてのDistributionではない。

## 2. コンパイラ的设计

前節でのべたように、現在のオペレーティング・システムにふくまれているFORTRAN IV コンパイラとIOCSに対してかなり改良の余地があることがわかったので、ある程度われわれ自身の手でやってみたいということになった。

まず言語の選択が問題になる。まずFORTRAN IVをやればよいようなものであるけれども、現在のFORTRAN IV言語を少し拡張してもっとゆとりのある——未熟な人でもエラーが少ないようなものがほしい、かなりFull ALGOLに近いものをとり入れたいということを考え、Static Storage AllocationでFORTRANと共用できるものをはじめに作ることにした。FORTRAN言語のシンタクスは構成があまりよくなく、ALGOLから出発した方が後の拡張に便利であるからである。この点については根田・浅井〔7〕を参照されたい。

次にオペレーティング・システムとの関連はあまり論じられてはいないけれども、実用上からは重要である。モニタをシステム・モニタとプロセッサ・モニタにわけると、計算センターとしてはシステム・モニタは種類がもっとも望ましく、種類をふやすことは極力さげねばならない。したがってわれわれのコンパイラは現在のシステム・モニタ（IBSYS）の下に入るものでなければならない。また現在用いられているプログラム言語との連関性も忘れてはならない。そうするとプロセッサ・モニタ（IBJOB）の管理下に入る必要もある。すなわち出力言語もきまってくる。I/O関係ではコンパイラ自身ではなるべくバッファリング・システムを利用し、出力のオブジェクト・プログラムでは必要のないIOCSの部分はコア常駐部からはずれるようにする。

コンパイラの構成は次のようになっている。

### (1) シンタクス・チェッカー

ストレージ・アロケータ

- (2) アリスメティック・トランスレータ
- (3) インデクス・アナライザとオブティマイザ
- (4) インストラクション・ジエネレータ
- (5) アセンブラ

これは構成の点では現在のFORTRAN M のものとほぼ同じであるが内容的にはかなりちがっている。(1)については根田・浅井〔7〕参照。(2)はFloyd〔6〕に近いもの。(3)はアルゴル・タイプであるとかかなりむつかしい。(4),(5)では現在MAP をつかっているのをはぶいてコンパイル終了時間をへらした。

全体として原研のI/Oユニットの低速性を考え、Phasing の手法を採用してある。すなわちデックごとくにコンパイルを完了させるのをやめ、全デックについて各パスを行なっていく。

またリスト処理を極力活用するようにした。

最後にこの仕事の副産物として、システム・プログラムを実際に作ってみるということは、現在使っているものの理解にもっとも役に立つということがある。単なるトレースに比べるとかなり本質的なものがつかめるようである。

## 文 献

- 〔1〕 Noble, Jr., A. S., Design of an Integrated Programming System and Operating System, Part1: System Considerations and the Monitor, IBM Systems Journal, June 1963, pp. 153-161
- 〔2〕 Talmadge, R. B., Part2: The Assembly Program and its language, ibid., pp. 162-179
- 〔3〕 Hedberg, R., Part3: The Expanded Function of the Loader, ibid. sept-Dec 1963, pp. 298-310
- 〔4〕 Lerner, R., Part4: The System's FORTRAN Compiler, ibid. pp. 311-321
- 〔5〕 White, B. and Trimble, J., Part6: Implementation on the 7040/44 Data Processing System, ibid. 1964, pp. 79-94
- 〔6〕 Floyd, R. W., An Algorithm for Coding Efficient Arithmetic operations, Comm. ACM. Jan. 1961, pp. 42-51
- 〔7〕 根田美佐子—浅井 清, シンタクス・チェッカーとストレージ・アロケータに  
おけるリスト・プロセッシング, 第7回プログラミング・シンポジウム報告集。

本 PDF ファイルは 1966 年発行の「第 7 回プログラミング・シンポジウム報告集」をスキャンし、項目ごとに整理して、情報処理学会電子図書館「情報学広場」に掲載するものです。

この出版物は情報処理学会への著作権譲渡がなされていませんが、情報処理学会公式 Web サイトに、下記「過去のプログラミング・シンポジウム報告集の利用許諾について」を掲載し、権利者の検索をおこないました。そのうえで同意をいただいたもの、お申し出のなかったものを掲載しています。

[https://www.ipsj.or.jp/topics/Past\\_reports.html](https://www.ipsj.or.jp/topics/Past_reports.html)

### 過去のプログラミング・シンポジウム報告集の利用許諾について

情報処理学会発行の出版物著作権は平成 12 年から情報処理学会著作権規程に従い、学会に帰属することになっています。

プログラミング・シンポジウムの報告集は、情報処理学会と設立の事情が異なるため、この改訂がシンポジウム内部で徹底しておらず、情報処理学会の他の出版物が情報学広場 (=情報処理学会電子図書館) で公開されているにも拘らず、古い報告集には公開されていないものが少からずありました。

プログラミング・シンポジウムは昭和 59 年に情報処理学会の一部門になりましたが、それ以前の報告集も含め、この度学会の他の出版物と同様の扱いにしたいと考えます。過去のすべての報告集の論文について、著作権者 (論文を執筆された故人の相続人) を探し出して利用許諾に関する同意を頂くことは困難ですので、一定期間の権利者搜索の努力をしたうえで、著作権者が見つからない場合も論文を情報学広場に掲載させていただきたいと思えます。その後、著作権者が発見され、情報学広場への掲載の継続に同意が得られなかった場合には、当該論文については、掲載を停止致します。

この措置にご意見のある方は、プログラミング・シンポジウムの辻尚史運営委員長 ([tsuji@math.s.chiba-u.ac.jp](mailto:tsuji@math.s.chiba-u.ac.jp)) までお申し出ください。

加えて、著作権者について情報をお持ちの方は事務局まで情報をお寄せくださいますようお願い申し上げます。

期間：2020 年 12 月 18 日 ~ 2021 年 3 月 19 日

掲載日：2020 年 12 月 18 日

プログラミング・シンポジウム委員会

情報処理学会著作権規程

<https://www.ipsj.or.jp/copyright/ronbun/copyright.html>