

## 怒 れ 怒 れ

—著作権について\*

西 村 恕 彦\*\*

会員諸氏はどうお考えだろうか。筆者は怒っているのである。情報産業は、将来に大きな利益を約束されているが、その一部は、情報の対価とか、プログラム権とかいう、未だ確立されていない商習慣へのあいまいな期待に基礎を置いている。著作権のように確立されている権利にたいする正当な評価でさえもがなされないような倫理感の風土の上に、どうして情報産業を築くことができようか。

筆者がこの何年間に、コボルについて書いた紹介や解説はかなりあるが、その一つ一つが、筆者にとってのステップであり、発見の過程であった。何か新しい視角や理解をひらくための努力であった。たとえばコボルでは、開発経過等に関する謝辞を必ず転載するように要求されている。筆者はこれを、毎回まったく独立に、新しく翻訳し直している。それは発見の努力であり、また一部分はその場にふさわしい文体を得るためである。こういう苦心と愛着とのある訳文が、たとえば、FACOM 230-10 COBOL、富士通、SP-071-1/1-3 のように、いとも無造作に、なんのためらいもなくそのまま丸写しにされているのを見ると、心が痛むのである。

あるいは、中川洋介：大型計算機による COBOL、数理科学 1968-9 の一部の記述<sup>1)</sup>は、筆者の理解、解釈以上の何物もつくわわっていない。いかに小さな一歩とはいえ、進歩が無くてよいわけはない。いやそれならばまだよい。悪くなっている例もある。大林久人：COBOL 演習、日本経営出版会 の用語解説<sup>2)</sup>は、筆者の原文が統一された思想と文体とで書かれているのにたいして、引き写しをしながら部分的な改変をしているので、同じことがらの説明が、場所によって不統一に変わっていて、そのうえ、まちがいを生じている。

筆者の解説では独自の基準によって項目をえらんで

いて、たとえば DIVIDE BY という動詞を項目として立てたのは、この書き方がポピュラーではないにも拘らず、理解しやすいという理由からであった。これを DIVIDE INTO と同じであると説明したのでは、まるで意味がないし、だいいち正反対のまちがいになっているのは、不勉強としかいえない。いったいこのひとは、コボル語の勉強をしたことがあるのだろうか。

紹介や解説はだれが書いても同じようなものになるだろうというのは表面的な見方である。何か独自のものがなければならぬのだ。まして、筆者はコボルの索引や用語解説についてはいささか工夫するところがあったので、こういう調子の流用、孫引きが、基礎的な勉強もなしに安直に利用されるのは、わびしいかぎりである。

もっとも、一步を進めた例もある。コボルで ALTER や DEFINE などの動詞は、コーディング例を示して説明することが望ましいにも拘らず、実用的な、本当らしい例がなかなか無かったのを、筆者が作成して教科書に使った。ところが、渡辺 茂ほか：COBOL によるプログラミング<sup>3)</sup>、日本事務能率協会 にこのコーディング例<sup>4)</sup>がそのまま丸写しにしてある。ただ筆者の原著では口頭で説明する趣旨で、何も注釈がつけてなかったのを、ていねいに説明してあるのは、読者にはいちおう親切であろう。「いちおう」といったのは筆者の例に、あるトリックが使っていたのがそのままになっているからである。

それにしても、かってに使って知らぬ顔という法はあるまい。コーディング例といえども、そっくり引き写しにすれば著作権の侵害になる。

ドキュメンテーションの良いプログラムで、出所がそれとわかるものが、何のことわりもなしに引用されているのを見ることがあるが、いったい原著者の許可を得ているのであろうか。

そのままの丸写しではなく、着想だけを拝借して、コーディングし直せばそれでかまわないだろうか。ま

\* 討論していただいた多くの方々に感謝します

\*\* 通商産業省工業技術院電気試験所

まったくかまわないのか、それとも、好ましくはないが抗議もできない、たんなる逃げ道なのであろうか。

こういう例はどうだろうか。魚田勝臣：MELCOM-1530 における COBOL とその適用業務例——水道料金計算，Computer Report 臨時増刊号 1966-10 にコーディング例が出ているが、その問題の設定と解析の主要部分は、筆者の教科書中の出題にそのままおんぶして、その解答（コーディング）は独自に作った形になっている。こういうことは差し支えないことなのだろうか。

逆に筆者自身はといえば、たとえば和田英一さんの書いたものはずいぶんよく読んでいて、その結果、着想や知識を無断で借用したことが何回もある（和田さん、ご免なさい）。しかし、少なくとも著作権の侵害となるような利用をしたことはないと信ずる。

偶然の一致ということもある。筆者が現在、準備中の教科書では、電気料金の計算を取り上げて説明してある。ところが先日、浦 昭二：FORTRAN 入門，培風館 をばらばら開いたら、まるきり同じ例題が使っているではないか。

これはまったく偶然の一致なのだが、読者（あるいは浦先生）の目には、模倣したうつるかもしれない（本や小説や番組の題名については、もはや模倣があたりまえの観がある）。

こういうコーディングや例題の流用は、一般の書物ではなく、メーカーの提供するマニュアル類についても起こることがある。たとえば日立のマニュアルに見られたまちがいは、それがクラリカルなものであるだけに、アイビーエムのマニュアルのまちがいを踏襲したのであろう。あるいは、日本アイビーエムで作った適性検査がレミントンで模倣されたということも、小さな誤植の一致から確認できたのである。

利用の形態が一般の著作物ではなくてメーカーの提供する資料であり、また利用される内容が一般の文章ではなく、例題や流れ図やコーディングである場合に、著作権等の保護をどう考えたらよいだろうか。

著作物ではなく、プログラムそのものときには、さらに困難な問題がある。

ソフトウェアは、開発したり、購入したりするよりも、格段に安い費用で手ばやく複製でき、しかも大きな価値を生み出すことができるという点で、その権利を有効に保護することが困難である。

システムやプログラムの開発者が、その権利の保有を主張することは容易であるが、それが侵害されない

ように有効に防衛することは困難である。たとえばこれが著作物であって、公刊物の形で侵害されたのであれば、それを発見することはむずかしくはない。公刊物の形ではなく、こっそりある部数を、制限された利用者のあいだ（たとえば講習会や委員会）で複製された場合には、あきらかに著作権等の侵害ではあっても、発見はほとんど不可能である。

こういう複製を実際的に抑制しているのは、多くの場合には、複製を作るよりも原本を買うほうが安くてはよいという理由だけであろう。だから複製を作ることが容易で、しかも安い場合には、これを有効に抑制することははなはだ困難であって、ソフトウェアの場合がまさにこれに相当する。似た困難は音楽の著作権にも見られる。たとえば喫茶店などで営利のために音楽を利用する者は、たんに楽譜やレコード盤の代価を払うだけで、手軽にそれを実行できる。そしてこれは一般にはあきらかに著作権の侵害であるにも拘らず、有効に発見することができない。

ある計算機のためのシステムやプログラムは、すべてそのメーカーが末端ユーザにたいして供給することが商習慣になっているから、このことは未だ深刻ではないが、計算機メーカー以外のソフトウェア製造業者が、末端ユーザにたいして直接にシステムを販売するようになると、この問題は表面化してくる。

つまりあるユーザがそのシステムを入手、利用しようと思ったら、ソフトウェア製造業者から高価に買うよりも、どこかのユーザのところへ、黙って複製してしまうほうが、はるかに手っとりばよい。このことは契約あるいは紳士協定として禁止することができるだけで、有効に阻止することはできないだろう。

機械語で書かれたプロセッサならばまだよい。共通性の高い言語で書かれたプロセッサとか、あるいはシステムの外部仕様とかは、どの計算機にも適用できる可能性がある。

たとえば筆者は、機械翻訳システムの完全なリステイング（共通性の高いコボルで書いてある）とそのドキュメンテーションを公開した。これが無断で利用されることの無いように筆者は希望しているが、実際問題としては、私企業が利益のためにこれを無断でコピーし実行してもこれを発見し有効に排除することができないのではないかと、筆者は悲観的なのである。

ソフトウェア等の権利の保護の問題は、ユーザのあいだばかりではなく、むしろメーカーのあいだでのほうが深刻だろう。

アイビーエム社のシステムは他の多くのメーカーが写して、かつたいていは何のトラブルも起きてはいない。この場合には、市場におけるアイビーエム社の優位性が、けっきょくは同社に利益を還元してくれるので、同社としてはむしろ写されること（普及すること）を歓迎する面もある。類似のことは、録音テープや写真フィルム等の規格や特許の公開にも見られる。

しかしアイビーエム社ですら、GPSS の存在とか、FORTRAN の超小型コンパイラの内容とかは、かなりの期間、秘密にしていたし、一般にプロセッサの流れ図やドキュメンテーションは、できるだけ秘匿するようにつとめている。

逆に、小さなメーカーが画期的なシステムを開発したとしてみよう。もしこれを他のメーカーが、とくにアイビーエム社のように圧倒的に優位に立つメーカーが、意識的に写して、大々的にセールスしたらどうなるであろうか。そのシステムから得られる利益の大部分は大メーカーが“横取り”してしまい、開発したメーカーのほ

うは利益のおこぼれを回してもらえるだけで、開発の努力に十分にむくわれるかどうかは疑わしい。

システムをかってに写すことの利害は、このように優位メーカーとそうでないメーカーとでは異なってくるが、写してもかまわないかどうかの規制を野放しにしておいてよいわけではない。システムを他のメーカーに開放するかどうかの決定権は、あきらかに当初の開発を行なったメーカーが優先的に保有すべきである。それが公正でもあり、そのような保護なしには、独創的な開発への意欲は抑制されてしまう。

いずれにせよ、無制限かつ不公正な模倣や侵害が起こらないようにする最初のとりでは、第一線の技術者や研究者の良心あるいは倫理でなければならぬ。

著作権のような確立されている権利さえも尊重できなくて、どうして情報産業の公正な発展が可能であろうか。筆者は怒っているのである。会員諸氏はどうお考えだろうか。

**付注** 参考までに、筆者の原文（左側）とそれを写したもの（右側）との一部の例を対照して示す。

### 1)

事務データ処理には順序ファイルの性質を利用したものが多く、その場合、予備処理としてレコードの分類（ならべかえ Sorting）が必要である。事務データ処理をきちんと記述するという目的からいっても、分類作業の記述の仕様を COBOL が含むことにははっきりした意義がある。

ファイルの再分類と多数回の通読とを含む、一連の長い事務データ処理のジョブをひとつのプログラムで統一的に記述できることになる。

### 2)

ABOUT 整数 RECORDS;  
 ファイルに含まれるレコードの件数  
 ACCEPT データ名;  
 少量のデータのための入力動詞  
 ACCESS MODE IS RANDOM;  
 ファイル管理文節での即時呼出しファイル指定  
 ACTUAL KEY IS データ名;  
 レコードの所在番地を指示する項目  
 ADD データ名 TO データ名;  
 たし算の動詞

### 3)

BETA SECTION. DEFINE jump WITH FORMAT  
 jump on *k* to *m*, *z*, *p*.  
 G. IF *k* IS LESS THAN ZERO GO TO *m* ELSE  
 IF = GO TO *z* ELSE IF GREATER GO TO *p*.  
 END DECLARATIVES.

事務データ処理 (BDP) の分野では、順序だったファイルの性質を利用した処理が多いものですが、その場合に予備的な処理としてレコードの分類が必要になってきます。

このようなことからいって、分類作業の記述に関する仕様とそれに関連する命令を COBOL が含むことは意義があると思います。ファイルの分類と多数回の通読とを含む一連の長いジョブをひとつのプログラムで統一的に記述できるということは、……

ABOUT 整数 RECORDS  
 ファイルに含まれるおよそのレコード数  
 ACCEPT データ名 FROM 装置名  
 少量のデータの入力動詞。ふつうはカード・リーダーから入力する  
 ACCESS MODE IS RANDOM  
 即時呼出しファイルの指定。FD 中の記述  
 ACTUAL KEY IS データ名  
 レコードの所在番地を示す項目  
 ADD データ名 TO データ名  
 加算の動詞

BETA SECTION. DEFINE JUMP WITH FORMAT JUMP ON  
K TO *M*, *Z*, *P*.  
 G. IF *K* IS LESS THAN ZERO GO TO *M* ELSE IF = GO TO  
*Z* ELSE IF GREATER  
 GO TO *P*.  
 END DECLARATIVES.

(昭和 43 年 9 月 25 日受付)