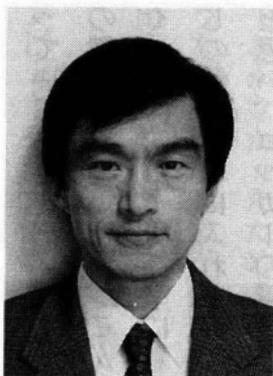


知の現在

# なぜコンピュータは間違えるか？

中 所 武 司



(ちゅうしよ・たけし) 明治大  
学理工学部情報科学科教授(ソフ  
トウェア工学)。一九四六年丸亀  
市生れ。東京大学工学部卒。同大  
学院修士課程修了。工学博士。著  
書「ソフトウェア工学」(朝倉書  
店)、「ソフトウェア危機とプログ  
ラミングパラダイム」(啓学出  
版)、「プログラミングツール」(昭  
晃堂、共著)、「人工知能」(同)  
など。

夜遅く、締め切りの迫った原稿を書き終え、翌朝、自宅から大学へ行く途中の郵便局の窓口で速達を頼んだときのことである。財布を忘れたことに気づき、あわてて封筒を取り返して駐車場の車に戻った。そして自宅へ引き返そうとしたとき、ふと郵便貯金のカードを持っていくことを思い出し、無事に速達を出すことができた。忘れた財布には大学の研究室の鍵もいれていたが、こんなときのために合鍵を鞆に入れていたので、そのまま大学にいくことにした。研究室に着いてから、机の引き出しから印鑑を出そうとしたとき、引き出しの鍵も財布に入れていたことを思い出した。さすがにこれはあきらめかけたが、もしやと思い、ロッカーの鍵を差し込むとこれで鍵が開いてしまったのである。

人間は予期せぬ出来事に会ったとき、その場で知恵を絞ってなんとか問題を解決しようとする。しかし、コンピュータはこうはいかない。彼らはときどき大事なときに間違いをしてかす困ったものたちである。まず、その事例をいくつか紹介しよう。

ある日、十年以上も正常に動いていたJ Rのみどりの窓口の端末が故障し、午前中停止してしまった。その原因は、あるタイプの端末を一八〇〇台から二二〇〇台に増設したとき、その端末を制御するプログラムが最大二〇四八台までしか扱うことができないように

作られていたにもかかわらず、コンピュータは自分の能力を過信して暴走したのである。

また別のある日、六年間正常に動作してきた航空管制システムがたまたまパイロットの異常に長いメッセージを受信したことが原因でダウンしてしまった。そのために旅客機の離発着ができなくなり、多くの乗客が空港で待たされてしまった。これもコンピュータが自分が処理できるデータの限界を知らずに暴走したためである。

これらの事例は、いずれも処理可能な最大値を越えたときの例外処理をコンピュータに教えておかなかったための事故である。通常はプログラムを作成するときにあらかじめこのような例外的な事象が発生したときの処理を記述しておくが、そのすべてをいつも事前に出出できるわけではない。そしてこの程度のコンピュータの事故は日常茶飯事なのである。

コンピュータが大惨事を引き起こすこともある。一九九四年の名古屋空港での航空機墜落事故では二百数十人が亡くなっているが、その原因の一つは、着陸やり直しの命令を受けたコンピュータが、その後のパイロットの着陸しようとする操作に反抗して異常な機首上げを引き起こしたためである。なんとコンピュータによる自動操縦とパイロットによる

手動操縦の両方の機能が同時に実行可能な設計になっていたのである。

さて、そろそろ本題に入ろう。「なぜコンピュータは間違えるか？」といえは、それは人間が間違えるからである。人間はコンピュータの頭脳にあたるプログラムの開発時に設計ミスをする。そしてそのミスを検出すべき検査ではテスト漏れにより見逃してしまう。かくしてコンピュータは人間が教えたとおりに動くが、人間が思ったとおりに動かない。結局、人間は自ら作りだしたコンピュータをコントロールできないのであり、コンピュータは自らをつくりだした人間を超えられないのである。

ではなぜ人間はプログラム開発において設計ミスやテスト漏れを回避できないのであるか。それは現在のプログラムの作り方に原因がある。実はコンピュータの実用化から今日までの半世紀の間、コンピュータへの命令を1ステップずつ手続的に記述するというプログラムの作り方はまったく変わっていないのである。

たとえば、銀行の端末からの預金の出し入れや送金を実行するシステムは、約一〇〇〇万ステップのプログラムからできている。これを一〇〇〇人のプログラマで開発したとすれば1人当たり一万ステップを作成したことになる。「データを読み込む」とか「残高と

引き出し額を比較する」とか「手数料を差し引く」などといった手続き処理を一〇〇〇万ステップも記述してミスがないほうが不思議である。

トーマス・クーンの「従来のパラダイムが行きづまり、危機的状況になったとき、新しいパラダイムが生まれ、科学革命が起こる」という科学史観を借りれば、現在はまさに高度情報化社会への激しい変化のなかで、この

手続き型のパラダイムに代わる新しいプログラミングパラダイムが求められている。パラダイムシフトのプロセスは下図に示すように6段階であらわせるが、プログラミングパ

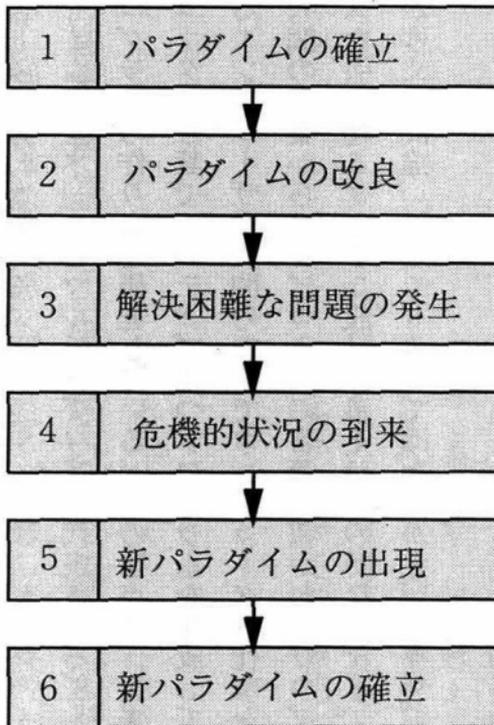


図 パラダイムシフトのプロセス

ラダタイムは第5段階での産みの苦しみを味わっている。

私は今、この問題に取り組み、インターネットとマルチメディアに代表される新しい情報時代にふさわしい新しいプログラムの作り方を研究している。その要点は、従来のように情報処理の専門家がコンピュータの言葉で1ステップずつプログラムを作成していくのではなく、業務の専門家であるエンドユーザが人間の言葉でコンピュータに仕事を教える方法をとることである。この人間の言葉からコンピュータの言葉で書かれたプログラムを自動生成してやれば、設計ミスを引き起こすような設計作業そのものがなくなり、コンピュータがコンピュータであるがゆえの間違いはなくせるのではないかという夢を追いかけている。

冒頭の問題指摘に戻るが、「人間は予期せぬ出来事を予期できるか？」という逆説的命題を乗り越えて、新しいパラダイムが確立する日まで、コンピュータは人間が教えたように間違うということをおぼえてはならない。