うつだより

第22号

編集発行人
清水吉男
(株)システム クリエイツ
横浜市緑区中山町869-9
電話 045-933-0379
FAX 045-931-9202

システム設計講座

テストとは元来退屈なもの、ましてや自分で書いたプログラムの場合は尚更である。誰でも自分の書いたプログラムは、自分のイメージした様に動くものと「願って」おり、その結果、一通りのテストで『完了』してしまうのです。そこでテストにメリハリをつけ、退屈しない工夫が必要なのです。

＾＾ 段階別テスト法とは？ ＾＾

テスト作業で退屈なのは、やはりテストの作戦を練ることでしょう。特に「単体テスト」段階での、各モジュールの「機能テスト」は、モジュールが正しく動くこと（？）を見せることに注意が払われ、その為に入力データも、大抵、正しく動くデータしか想定されません。そして「うまく動くことを見届けた」所で、なぜか「機能テスト」は『合格』となってしまいます。

現実にこれ以上は、各人の意志に委ねられているのです。「段階別テスト法」は、この様に個人のスキルに全面的に依存することなく、自分の書いたプログラムを客観的にテストするためのガイドラインを与えるものとして提唱されたもので、現在では次の様に五種類の段階を想定しています。

一、機能テスト段階
二、入力テスト段階
三、出力テスト段階
四、相互機能テスト段階
五、コーディング・テスト段階

この内、「機能テスト段階」と「コーディング・テスト段階」は、所謂「単体テスト」に用いられるもので、その他のテスト段階は、「単体テスト」と「統合テスト」（インターフェイス・テスト）の両方に用いられます。

＾＾ 機能テスト段階 ＾＾

これは、各モジュールに定められた「機能」をテストするもので、「仕様」に定められた「機能」が予定通りに『機能すること』を確かめることが、この段階での中心的な作業となります。このテスト段階は「単体テスト」に於て実施されることを想定しており、テストは単一のモジュールか、それを取り巻く幾つかのモジュールを組み合わせた程度で実施されます。

勿論モジュールによって、データを変換することが「機能」であったり、チェックすることが「機能」であったりします。したがって、そこに定められた「機能」に適したテスト・ケースを想定する必要があります。が一般に次の点に焦点を合わせます。

（一）「機能」を実現するための入力データと、その時の出力結果を想定したもの。
（二）最低限の「不正な」入力データ。

この結果、この段階のテストを通過することによって、先ずは「ブラックボックス」となります。しかしながらこれは「取り敢えず動く」ということが分かっただけです。

にも関わらず、現実はこの程度で「単体テスト」を「合格」しているのです。

それでも上記の視点からテスト・ケースをしっかりと用意している場合は、この段階のテストだけでかなりのテストが行なえるでしょうが、多くの場合は、事前に十分なテスト・ケースを想定し、吟味していることはありません。それだけに、この段階のテストで終わらずに、視点を変えた他のテストを併せて実施すべきなのです。

＾＾ 入力テスト段階及び出力テスト段階 ＾＾

これは「機能テスト段階」で定めた入出力データのテスト・ケースを更に拡張して実施するテスト段階で、目的は「モジュールに定められた機能が正常に働く範囲を確かめる」ことです。従って、「限界値テスト」や「同値分割」によるテストは「入力テスト段階」として実施されます。また出力に関しても、出力レコードのフォーマットやメッセージ等は、ただ漠然とテストを行なうのではなく、エラー・メッセージ的を絞ってテストすべきでしょう。勿論この場合も全ての組み合わせをチェックすることは事実上不可能です。

＾＾ 相互機能テスト段階 ＾＾

これはインターフェイスをテストする段階です。一般に個々のモジュールに対する仕様は定めていても、相互作用については明記されていることは少なく、このテストを行なう為には、全体を記述した仕様や構造図が必要です。

＾＾ コーディング・テスト段階 ＾＾

これはモジュールの内部構造を網羅したテストで、所謂「論理網羅テスト」と呼ばれるテストに相当します。このテストは場合によってはツールの助けを必要とするかも知れません。またテストの対象は個々のモジュール単位で、「単体テスト」の様な形になりますが、必ずしも「統合テスト」の前に終了している必要はなく、担当モジュールを「統合テスト」に提供したあとで、並行して実施すべき性質のテストです。

＾＾ 総括 ＾＾

一般に自分で書いたプログラムを自分でテストするのは容易ではありません。多くの人は「自分の最初の行為」を否定することに抵抗するし、その様な「否定」に対しては「弁護」するものです。しかしながら「テスト」とは『自己を否定』することでもあり、「弁護士」と「検事」を一人で演じるようなものです。したがって、この「二面性」をコントロール出来なければ、良いテストは出来ないことになってしまいます。

「段階別テスト法」によって、個人のスキルに依存する割合は減少したとしても、テスト・ケースを想定する段階では、やはり「検事」であることが望ましいでしょう。●

原稿量の関係で、今回試みに文字を1ポイント小さくしてみました。

IBMのゆくえ

▼IBMがダウンサイジングの波をまともに被っている。二〇年前に発売した360シリーズを皮切りに、今日迄世界のコンピュータ市場を制覇したIBMが、その巨大さ故に、まるでマンモスの様に急速な環境の変化について行けずにあえいでいる。

▼日本に於けるダウンサイジングの兆候は、丁度一年前の四月の統計に現われ始めたことを、昨年の十二月号のこの稿で取り上げたが、この年日本IBMは売上高の七三%をたった二七〇余社から稼いでいたのである。そして米国IBMが今年の第一四半期が赤字であったことを五月二七日に公表し、同時に日本IBMも前年同期の売上を二五%も下回ったことを発表した。

▼メモリーの密度は二年で四倍と急速に向上し、今や人差し指の先に数百万もの文字が入る。また手の平に収まる程度のCPUも、その能力は一般の汎用機を凌ぐところまで進んできた。この世界の過去一〇年間の進歩は、まさに驚異としか形容できない。

▼ダウンサイジングは現実の問題となって姿を現した。それも一気に目の前に現われたのである。国内の汎用機メーカーも我が事として動き始めるだろう。利益率の高い大型機販売に馴れ親しんだ組織が、どれだけ素早くダウンサイジングに対応するか見物である。

暁鐘の音

真実は全体にあり（二）

数ヵ月にボーイングの旅客機が、逆噴射が原因で墜落した事故があった。エンジンが飛行中に逆噴射するなど、関係者に言わせれば「ありえないこと」なのである（もっとも、あり得るのでは困るが）。確かに「逆噴射」に関しては何重にもガードがかかっている思われる。だが現実に逆噴射し、多くの人命が失われたという事実がある。

旅客機の様な巨大なシステムになると、全体を把握することは極めて困難であろうことは推測がつく。現在の技術では、幾つかの部分に分け、その中で完成度を上げるしかないかもしれない。しかしながら『物』として機能するのは、それらの「部分」が集まって一つになった『物』である。

§　§　§　§　§

医学の世界も細かな「専門分野」分けられ、その結果、患者の特定の「部分」については意見を述べることはできるが他の部分や患者そのものに関する意見が出せなくなっているという。例えば胃腸を専門とする消化器系の医師は、その原因が「歯」や「顎」にあることを必ずしも指摘できない。もっと深刻な問題として、「癌」に侵された患者に対して、自分の専門外の部位に転移している場合、見過ごしてしまう事態も起きている。明らかに『人』を診れる医者がいなくなっているのである。

§　§　§　§　§

企業の世界でも同じことが起きている。企業の最高意志決定機関とし、例えば常務会などがあるが、各役員が各々「担当」部門を持っているために、他部門の情報を得ることが出来ないのと、得なくても済む（？）ことから、他の役員が出した案件に対して意見を出すことが出来ず、議論そのものが成り立たない状況に陥っているのである。つまりこの場合、「常務」は「部長」の単なる延長線上にあり、企業の全般を観るようにはなっていないのである。当然、その企業において時をおかずに弊害がでることは言うまでもない。

このように現実は「全体」を見ることに対する意識が低く、その為に社会全体が多大な損失を被っている。

§　§　§　§　§

残念ながらソフトウェアの世界にもこれがあてはまる。現実にシステムが大きくなったことに対する対応は、（一）単にシステムを対処可能な一定の大きさに切り、（二）「部分」の数に見合う人数を手当てし、（三）一つづつ割り当てるだけである。全体を見るために必要で且つ有効な努力は殆ど行なわれていない。

§　§　§　§　§

ソフトウェア・エンジニアリングは、なんとかして「全体」を把握しようとする方向にあり、システムが大きくなるにつれて、「設計法」や「分析法」というものが考えられてきた。設計法でいうところの『構造図』は接続するモジュールの全体像を捉え、それらのモジュール間の『関わりあい方』を見せることを目的としている。個々のモジュールは正しく（？）作られていても、それらの関わり方、関わらせ方を間違うとシステムは機能しないことを、関係者は何度も経験してきた筈である。

それでも、もっと大きなシステムになると、『構造図』だけでは収拾が付かない。そこで考えられたのが「分析法」である。そこではシステムにおいて「何が考えられているか」、逆に「何が考えられていないか」、そして「一体このシステムに何が起こりうるか」と言ったことを考えるための手立てを与えることである。

◎　—　◎　—　◎

この様な方法論は、単に新しい手法を「導入する／しない」の問題ではなく、如何にして『全体を捉えるか』と言う立場で考えるべきである。

◆

今月の一言

「与（とも）に共に学ぶべきも、未だ与に道に適（ゆ）くべからず。与に道を適くべきも、未だ与に立つべからず。与に立つべきも、未だ与に権（はか）るべからず」

論語（子罕）

◇—◇—◇—◇—◇

一緒に勉強するということはそれ程難しくはないが、一緒に人生を歩むとなると容易ではない。よしんば共に行くことが出来ても、共に（権力の）地位につくことは出来ない、と孔子は言うのである。

夫婦であり続けることも、共に事業を起こし、創業から守成へと関係を維持し続けることも、共に道を行くことであり、その人の思想や生き方等が深く絡み合うために幾多の困難を伴う。だから「道友」は「知己」でなければ成り立たない。

それでも共に権ろうとしない限り、この関係は何とか維持できる。「権」は秤の分銅のことで、誰も不満のないように釣り合いを権（はか）らなければならない。したがって共に権ろうとすれば、必ず混乱が起きる。

先ごろアスキーの創業者達が袂を分かった。創業の段階では多少の問題があっても、「目標」の前に影を潜めるが、守成の段階に入れば情勢は違ってくる。何時の時代にあってもやはり「与に権るべからず」である。